UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

MANAGING COMPUTATIONAL COMPLEXITY THROUGH USING PARTITIONING, APPROXIMATION AND COORDINATION

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

REZA ALIZADEH KORDABAD

Norman, Oklahoma

2022

Downloaded from SHAREOK. Persistent Handle URL: https://hdl.handle.net/11244/336278

MANAGING COMPUTATIONAL COMPLEXITY THROUGH USING PARTITIONING, APPROXIMATION AND COORDINATION

A DISSERTATION APPROVED FOR THE

SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY THE COMMITTEE CONSISTING OF

Professor Janet K. Allen, Co-chair

Professor Farrokh Mistree, Co-chair

Dr. Charles Nicholson

Dr. Thomas Neeson

Dr. Shabnam Rezapour

© Copyright by REZA ALIZADEH KORDABAD 2022

All Rights Reserved.

Ağacan, my hero who inspired me to become the best version of myself

Anam, my safe heaven, who has sacrificed everything for me

Leylim, my best friend, ala gözlüm, the head of my support system, but most importantly, my love

Qanadsız Mələyim, my angel without wings

Dadaşım, my hero who inspired me to become an achiever

Muşabehim, my mentor who taught me how to read and write

Sevilim and Yaşarım, my light in the darkest in the darkest moments

Acknowledgements

This dissertation is a result of the continuous support, mentoring and encouragement from several people involved in my life. The associations with these people have helped me grow as a person and as a professional. They have played a huge role in helping me in my journey to realize myself. I express my gratitude to each of them and acknowledge them here.

I begin by thanking the two most influential people of my life during my doctoral journey – my academic parents. I thank my Ph.D. advisors Professors Janet K. Allen and Farrokh Mistree for their continuous encouragement, support, and faith in me. Their passion to help students realize their full potential by providing them an opportunity to learn is truly amazing. Their dreams for me are beyond what I dream for myself. They have helped me in defining long-term and short-term goals and helped me to realize these goals through their support and mentoring. They have always challenged me to go beyond my comfort zone. Through this experience I realized that it is fun to step out of one's comfort zone. The fundamental principles embodied within the Systems Realization Laboratory that is co-directed by Janet and Farrokh is to focus on scholarship and critical thinking. These principles have impacted me a lot and I plan to abide to these when I start my own research laboratory in future. As I reflect upon myself, I realize that I have significantly grown as a person both professionally and personally in the last four years with Janet and Farrokh.

I thank my Ph.D. dissertation committee members Dr. Charles Nicholson, Dr. Thomas Neeson, Dr. Shabnam Rezapour, Professor Janet K. Allen, and Professor Farrokh Mistree for critiquing my doctoral work and providing their insightful comments and feedback. I thank Dr. Nicholson and Dr. Rezapour for their comments during my Ph.D. General Exam which helped me frame the problem properly leading to my first two journal papers. I also thank Dr. Neeson for the encouraging and supporting words that he always has for me. I also thank Dr. Rezapour for her

comments on my comprehensive problem in Chapter 8. The advice she gave for my work, the material she shared with me, valuable guidance, help and support she gave me during the meetings we have had, the material she shared with me for a successful future career as a professor are very valuable and I thank her for that. I also thank Dr. Rezapour for her support without which I would not be able to have the opportunity to join the Systems Realization Laboratory. I am also very grateful for the financial support from the John and Mary Moore Chair, and the L.A. Comp Chair at the University of Oklahoma.

I thank all the colleagues with whom I worked during my doctoral studies. I thank Liangyue Jia from Beijing Institute of Technology that I had the opportunity of mentoring him and publishing two journal papers with him entitled “Ensemble of surrogates and cross-validation for rapid and accurate predictions using small data sets” and “A rule-based method for automated surrogate model selection”, and chapters 6 and 7 of my dissertation. I also thank Professor Guoxin Wang and Dr. Jia Hao from Beijing Institute of Technology. Moreover, I thank Sara Hajihashemi, ISE master student who I mentored and published paper entitled “Impact of Asset Management in a Green Supply Chain”, and Chapter 6. Furthermore, I appreciate Jack Williams, undergraduate HERE scholar whom I mentored and published “Using network partitioning to design a green supply chain”, and Chapter 6.

I thank all the mentors I have had during my academic life. I thank Professor Peter D. Lund from Aalto University for his continuous support, help, advice, and mentorship since 2014. I thank my brother-in-law, Dr. Khosrov Dabbagh Sadeghipour for his continuous support, help, advice, and mentorship. I thank Professor Gerard Cachon, for his comments on my green supply chain project.

I thank all the members of the Systems Realization Laboratory @ OU (starting Spring 2017) who were next to me, inspired and helped me during the last four years especially Dr. Shabnam Rezapour, Dr. Anand Balu Nellippallil, Pranav Mohan, Gehendra Sharma, Abhishek Yadav, Sara Hajihashemi, Jack Williams, Jackson Autrey, Mary (McCarty) Wallace, Dr. Zhenjun Ming, Maryam Sabeghi, Dr. Jelena Milisavljevic, Dr. Ru Wang, Xiwen Shang, Shan Peng, Meghnath Reddy Challa, Professor Guoxin Wang, Jackson Autrey, Ali Shahbazi, Shuting Chen, Dillan Portillo, and Shehnaz Sheik. I thank the staff members of Schools of ISE and AME especially Cheryl Carney, Kristi Huckabee-Wilson, Melodi Franklin, Melissa Foster, Bethany Burklund and Ellen McKenzie for all their help in these two schools. I thank Dr. Reza Maknoon for mentoring me during my masters and supporting me towards my doctoral studies with Janet and Farrokh. I wouldn’t be here without his support.

Acknowledgements ..... **v**

List of Figures ..... **xxiii**

Summary ..... **xxvii**

Chapter 1 Motivation and Problem Identification **1**

1.1 Motivation for Managing Computational Complexity and Uncertainty in Designing Evolving CPS Systems 3

1.1.1. Motivation of designing multi-echelon, multi-channel supply chain design in a low carbon, push-pull economy *4*

1.1.2. Addressing the challenges of managing computational complexity and uncertainty in multi-echelon, multi-channel supply chain *5*

1.2 Gaps, Research Questions/Questions, Hypothesis, New Knowledge, Functionality, and Utility 6

1.3 Validation Square 10

1.3.1. Theoretical structural validity – Is the design method internally consistent? *10*

1.3.2. Empirical structural validity – Are the example problems used in modeling the method appropriate choices? *10*

1.3.3. Empirical performance validity – Does the application of the method to the problems in the dissertation produce practical results? *11*

1.3.4. Theoretical performance validity – Does the application of the method to other problems produce practical results? *11*

1.4 Contributions 11

1.5 Research Thrusts and Layout of the Dissertation 13

1.6 Verification and Validation of Chapters of Dissertation 14

1.7 Dissertation Overview and Roadmap 16

Chapter 2 Frame Of Refence and Literature Review 19

2.1 Managing Computational Complexity and Uncertainty in Designing Multi-Echelon, Multi-Channel Supply Chain 20

2.1.1. Motivation of designing multi-echelon, multi-channel supply chain design in a low carbon, push-pull economy 21

2.1.2. Addressing the challenges of managing computational complexity and uncertainty in designing multi-echelon, multi-channel supply chain 22

2.2 Creating Partitioning-Approximation-Coordination Design Approach 22

2.2.1. Problem statement of managing computational complexity using partitioning and coordination 24

2.2.2. Using network partitioning to manage the computational complexity 25

2.3 Partitioning-Based Decision-Making Approach (T1 in Figure 2.2).....25

2.4 Approximation of Reality Using Surrogate Modeling (T2 in Figure 2.1) 28

2.4.1. Finding appropriate criteria for surrogate model selection (T2.1 in Figure 2.1) 28

2.4.2. Using time dependent features in surrogate modeling (T2.2 in Figure 2.1) 29

2.4.3. Using machine learning to automatically find appropriate function types and hyper-parameters *29*

2.4.4. Using ensemble of surrogates to use the advantages of various individual surrogates when the sample is undersized (T2.4 in Figure 2.1) *30*

2.4.5. Automating the process of surrogate modeling (T4 in Figure 2.1) *30*

2.5 Using Network Partitioning to Design a Green Supply Chain 30

2.6 Role of Chapter 2 in This Dissertation 32

Chapter 3 Surrogate Model Classification and Selection Criteria ..33

3.1 Critical Evaluation of The Review Papers Written on Surrogate Modeling ..37

3.2 Design of Experiments (DOEs) 51

3.2.1. The problem of size in surrogate modeling *51*

3.2.2. Principal component analysis as a dimensionality reduction method *52*

3.2.1. Variable screening as a dimensionality reduction method *52*

3.2.2. Partitioning as a dimensionality reduction method *53*

3.2.3. Bayesian updating as a dimensionality reduction method *53*

3.2.4. The main types of experimental designs *55*

3.2.5. Factorial designs *55*

3.2.6. Fractional factorial designs (FFDs) *56*

3.2.7. Central composite designs *56*

3.2.8. Orthogonal arrays *57*

3.2.9. Space-filling designs *57*

3.3 Comparison of Different DOEs: Evaluation Metrics 57

3.3.1. Unsaturated, Saturated, and Supersaturated Designs *58*

3.3.2. Minimum bias *59*

3.3.3. Minimum variance *59*

3.3.4. Rotatability *59*

3.3.5. Orthogonality *60*

3.4 Model Choice in Surrogate Modeling 61

3.4.1. Response surfaces *61*

3.4.2. Kriging *62*

3.4.3. Radial basis functions (RBF) *63*

3.4.4. Inductive learning *63*

3.4.5. Boosted trees *64*

3.4.6. Random forests *66*

3.4.7. Adaptive learning/active learning *68*

3.4.8. Hyperdimensional performance models *68*

3.4.9. Multiple surrogates *70*

3.5 Model Fitting in Surrogate Modeling 75

3.5.1. Least squares analysis to create the surrogate model *75*

3.5.2. Weighted least square regression (WLSR) *76*

3.5.3. Log-likelihood *76*

3.5.4. Backpropagation *77*

3.5.5. R-square (R^2) *77*

3.5.6. Mean absolute percentage error (MAPE) *78*

3.5.7. Mean absolute error (MAE) *78*

3.5.8. Cross-validation *79*

3.6 Comparison of Chosen Surrogate Models 81

3.7 Results of Critical Evaluation of Literature on Surrogate Modeling 82

3.7.1. Review of the applications of DOE, data representation models, and model-fitting methods *83*

3.7.2. Suggestions for DOEs, model selection and application, and model-fitting methods *85*

3.8 Verification and Validation of Critical Evaluation of Literature on Surrogate Modeling 96

3.8.1. Individual model verification and validation *96*

3.8.2. Multiscale model verification and validation *97*

3.8.3. Design process verification and validation *98*

3.8.4. Design outcome verification and validation *99*

3.8.5. Verification and validation in this dissertation *100*

3.9 Closing Remarks on Critical Evaluation of the Surrogate Modeling Literature 101

Chapter 4 Approximation of Reality Using Temporal Surrogate Modeling 104

4.1 Frame of Reference on Building Temporal Surrogate Models 106

4.2 Red River Basin and Available Data 114

4.3 Method for Predicting Water Inflow in Buffalo Reservoir 117

4.4 Predicting the Inflow in Buffalo Reservoir using Multivariate AutoRegressive Integrated Moving Average (MARIMA) 122

4.5 Predicting the Inflow in Buffalo Reservoir Using RF 124

4.6 Predicting the Inflow in Buffalo Reservoir using Artificial Neural Network (ANN) 126

4.7 Results and Discussions of Predicting the Inflow in Buffalo Reservoir 128

4.8 On Verification and Validation – Theoretical Structural Validity (TSV) 133

4.8.1. TSV of function-based design (Gap 4 in Figure 4.7) 134

4.8.2. TSV of temporal surrogate models (Gap 4 in Figure 4.7) 135

4.9 Closing Remarks on Predicting the Inflow in Buffalo Reservoir 136

Chapter 5 Approximation of Reality Using Spatial Surrogate Modeling 139

5.1 Frame of Reference for Spatial Surrogate Models 140

5.2 Predictors and Dependence in Space of Crime Rates 142

5.2.1. Predictors used in crime rate prediction in Los Angeles city..... 142

5.2.2. Spatial dependence of crime data..... 143

5.2.3. Critical evaluation of the literature on crime research in Los Angeles 144

5.3 Data Set Used for Crime Rate Prediction in Los Angeles City.....146

5.3.1. Describing the data, features, and characteristics..... 146

5.4 Methods Used for Crime Rate Prediction and Different Spatial Analysis...147

5.4.1. Point pattern analysis method to spatially analysis the location under study 148

5.4.2. Variance-mean ratio (VMR) analysis of crime rate prediction in Los Angeles city . 149

5.4.3. Distance-based point pattern measures 149

5.4.4. Nearest neighbor distances 149

5.4.5. Implementing the G function over the data set..... 150

5.4.6. Implementing the F function over the data set 151

5.4.7. Implementing the K Function over the data set..... 151

5.4.8. Interpretation of the results of hypothesis testing for F, G, K..... 151

5.5 Spatial Prediction Model 152

5.5.1. Step 1: Linear spatial regression model..... 154

5.5.2. Step 2: Identifying the spatial autocorrelation..... 154

5.5.3. Step 3: building geographically weighted regression model 155

5.6 Results and Discussion of Spatial Prediction Model.....156

5.6.1. Visualization of the distribution of the crime data over the Los Angeles city 156

5.6.2. Results and discussion of fitting OLS linear regression 158

5.6.3. Result and discussion of fitting the GWR model..... 160

5.6.4. Results and discussion of comparison between OLS and GWR models. 161

5.6.5. Results and discussion of Clark-Evans R statistics analysis 162

5.6.6. Results and discussion of G, F, and K statistics analysis 162

5.7 On Verification and Validation – Empirical Structural and Performance Validation.....163

5.8 Closing Remarks on Predicting Crime Rate using Spatial Surrogate Models 164

Chapter 6 Creating Mathematics for Building Ensemble of Surrogate Models 166

6.1 Frame of Reference on Building Ensemble of Surrogate Models.....168

6.2 Predicting the Microstructure of the Final Rod in a Hot Rod Rolling Problem 176

6.3 Method for Selecting the Surrogate Model Based on Time, Size, and Accuracy 181

6.3.1. Surrogate modeling process 181

6.3.2. Design of experiments and cross-validation 183

6.3.3. Function fitting using different surrogate models 183

6.3.4. Fitting a response surface model to the dataset 184

6.3.5. Fitting a Kriging model to the dataset 184

6.3.6. Fitting a radial basis function model to the dataset 185

6.3.7. Creating an ensemble of surrogate models using different surrogate models 186

6.3.8. Prediction metric: root mean square error 186

6.4 Results and Discussion on Hot Rod Rolling Problem 187

6.4.1. Comparison between ensembles of surrogates and individual surrogates 187

6.4.2. Trade-offs among accuracy, size and time 189

6.4.3. Results for output Y1 (allotriomorphic ferrite) 192

6.4.4. Results for output Y2 (Widmanstätten ferrite) 194

6.4.1. Results for output Y3 (pearlite) 195

6.5 On Verification and Validation 201

6.5.1. Empirical structural validation 201

6.5.2. Empirical performance validation 201

6.6 Closing Remarks on Building Ensemble of Surrogate Models 202

Chapter 7 Creating an Automatic Multi-Level Surrogate Model Selection Process 204

7.1 Frame of Reference of Surrogate Model Selection Process 207

7.1.1. Manual comparison based surrogate model selection 211

7.1.2. Evolutionary algorithm based surrogate model selection 214

7.2 Surrogate Model Selection Problem 219

7.2.1. Three-layer structure of surrogate model selection 219

7.2.2. Metrics for performance measures 221

7.3 AUTOSM Framework 222

7.3.1. Feature extractor 223

7.3.2. Surrogate model evaluator 225

7.3.3. Model type selector 225

7.4 Verification of AUTOSM 227

7.4.1. Surrogate model pool 227

7.4.2. Theoretical problems 228

7.4.3. Test problems used to demonstrate the utility of the proposed framework 230

7.5 Results and Discussion on Creating an Automatic Multi-Level Surrogate Model Selection Process 236

7.6 On Verification and Validation Creating an Automatic Multi-Level Surrogate Model Selection Process 247

7.6.1. Theoretical Structural Validation 247

7.7 Closing Remarks on Creating an Automatic Multi-Level Surrogate Model Selection Process 248

Chapter 8 Designing Evolving CPS Systems Transitioning from Push to Pull Low Carbon Economy 250

8.1 Frame of Reference on Green Supply Chain Design 250

8.2 Modeling the Green Supply Chain Problem as a Two-Echelon Supply Chain 261

8.2.1. Modeling decision interactions 261

8.2.2. Problem formulation of green supply chain design as a retail store density problem and a linear program..... 264

8.2.3. Solving the designed green supply chain and the retail store density problem 268

8.2.4. Results of solving the designed green supply chain and the retail store density problem..... 271

8.2.5. Take aways of solving the designed green supply chain and the retail store density problem..... 283

8.3 Impact of Asset Management in a Green Supply Chain 285

8.3.1. Frame of reference on asset management in a green supply chain 286

8.3.2. Method used for asset management in a green supply chain..... 290

8.3.3. Proposed mathematical model to solve asset management problem in a GSC..... 298

8.3.4. Results and discussion on the solution of asset management problem in a GSC..... 300

8.3.5. Verification and validation on the solution of asset management problem in a GSC309

8.3.6. Take aways of solving the asset management problem in a GSC 311

8.4 Design A Bi-Level Programming Model for Two Channels in A Three-Echelon Supply Chain 314

8.4.1. Mathematical model for multi-channel (online and traditional shopping), multi-echelon (warehouse, store, and customer) green supply chain..... 315

8.4.2. Surrogate modeling to approximate the lower-level objective function..... 324

8.4.3. Surrogate-based approximation algorithm (partitioning – approximation - coordination framework)..... 326

8.4.4. Results and discussion on solving the comprehensive problem of two channels in a three-echelon supply chain design problem 334

8.4.5. Scenario analysis on different percentages of online shoppers in a two-channel, three-echelon supply chain 344

8.4.6. Verification and validation of solving the comprehensive problem of two channels in a three-echelon supply chain design problem 348

8.4.7. Take aways on solving the comprehensive problem of two channels in a three-echelon supply chain design problem 349

Chapter 9 Closing Remarks, I statement and Way Forward, I Statement.....352

9.1 Problem and Assumptions 352

9.2 Contributions 354

9.3 I Statement 359

9.4 Research Thrusts and Applications 361

9.4.1. RT 1. What is the mathematics underlying automating the data-driven modeling and approximation models? 362

9.4.2. RT 2. What is the mathematics to capture the dimensions of an evolving CPS system and the dynamics among them?..... 365

9.4.3. RT 3. Data-driven, mathematical modeling for energy economics and climate policy in complex energy systems..... 365

9.4.4. RT 4. Blockchain applications in healthcare, energy, supply chain and other CPS systems..... 366

9.4.5. RT 5. Social aspect of the cyber-physical-social systems..... 367

9.4.6. RT 6. Environmental sustainability in the cyber-physical-social systems..... 368

9.4.1. RT 7. Green supply chain as a cyber-physical-social system using digital threads . 369

9.5 Overall Theme of my Research based on my PhD.....373

Appendices 375

List of Tables

Table 1.1. Verification types and their associated chapters 15
Table 1.2. Connection between research questions, chapters, and validation square 16
Table 3.1. Questions investigated in this chapter. 38
Table 3.2. Summary of previous review papers 45
Table 3.3. Surrogate modeling techniques. 83
Table 3.4. Critical literature review table: survey of engineering application of DOE, data representation models, and model fitting. 89
Table 3.5. General characteristics of surrogate models 93
Table 4.1. Data description 118
Table 4.3. Performance evaluation and comparison between MARIMA, ANN and RF models. 129
Table 4.4. Accuracy of the used forecast method in various forecast time horizons 130
Table 5.1. Results of fitting the spatial linear regression model. 158
Table 5.2. Results of Moran's I test using K nearest neighbors' method. 160
Table 5.3. Comparison between OLS and GWR models. 161
Table 6.1. Critical evaluation of the ensemble of surrogates' literature. 170
Table 6.2. Trade-offs among three criteria. 175
Table 6.3. The design variables 179
Table 6.4. Factors and factor levels for DOE 183
Table 6.5. Statistical analysis of RMSE value by various surrogates in 10 runs. 189
Table 6.6. Program run time to compute values for four surrogates from 9 to2-fold training data for output Y1. 191
Table 6.7. RMS errors generated by four different surrogates from 9-2 fold training data for output Y1. 193
Table 6.8. RMS errors generated by four different surrogates from 9-2 fold train data for output Y2. 195
Table 6.9. RMS errors generated by four different surrogates from 9-2 fold train data for output Y3. 196
Table 6.10. RMS errors generated by four surrogates from 9-2 fold training data for output Y1, Y2, and Y3. 197
Table 6.11. Specific guidance for the selection of surrogate models for output Y1 (Allotriomorphic Ferrite) 199
Table 7.1. Critical evaluation of the literature and gap identification. 215

Table 7.2. Surrogate-optional types and their corresponding hyper-parameters in the surrogate model pool 227

Table 7.3. Meta-features and the corresponding best surrogate optional type for theoretical problems 229

Table 7.4. Design variables and their definition of hot rod rolling design 233

Table 7.5. Meta-features and the corresponding best surrogate optional type for hot rod rolling and blowpipe design problems 235

Table 7.6. Average predicted accuracy for benchmark functions. 237

Table 7.7. The set of promising surrogate models (if the Best Type is one of the three CSOTs, bold the corresponding item) 237

Table 7.8. Comparisons between our work and other studies in surrogate model prediction 240

Table 7.9. The surrogate model type prediction running time of COSMOS and AutoSM. 244

Table 7.10. Robustness of surrogate model with surrogate types and surrogate optional types 244

Table 8.1. Summary of the critical evaluation of the literature on GSC design problem. 260

Table 8.2. Parameter estimations 271

Table 8.3. Cost functions applied to parameter estimations 272

Table 8.4. Carbon tax gap reduction. 278

Table 8.5. Operating cost and emission values. 279

Table 8.6. Different scenarios 280

Table 8.7. Different scenarios. 310

Table 8.8. Different weight factors 310

Table 8.9. Two definitions for the mapping function 325

Table 8.10. Solution for the upper-level function 338

List of Figures

Figure 1.1. Partitioning-approximation-coordination framework. 13

Figure 1.2. Skeleton of the dissertation. 14

Figure 1.3. Organization of Dissertation Chapters according to Verification and Validation Square. 15

Figure 1.4. The dissertation overview and roadmap. 17

Figure 2.1. Relationship of research efforts with the constructs of the systems-based design architecture and connection between chapters of the dissertation. 19

Figure 2.2. The idea of partitioning and coordination. 23

Figure 2.3. A partition hierarchy of a complex system. 26

Figure 2.4. Process for implementing partitioning-based design. 27

Figure 2.5. Finding coupling approaches between partitioning and coordination decisions 28

Figure 3.1. Relationship of research efforts with the constructs of the systems-based design architecture and connection between chapters of the dissertation. 34

Figure 3.2. Comparison between two experiments in an engineering design problem. 79

Figure 3.3. Problem characterization based on the two aspects of time and size. 85

Figure 3.4. Time-size-accuracy triangle 86

Figure 3.5. The Classification of some surrogate modeling techniques in a time-size-accuracy triangle. 94

Figure 3.6. Model verification and validation process (Sargent, 2010). 97

Figure 3.7. The verification and validation square (Pedersen et al., 2000; Seepersad et al., 2006). 99

Figure 4.1. Surrogate modeling in dam flow measurement. 110

Figure 4.2. Positions of the 38 dams over the red river basin (Zamani Sabzi et al., 2019). 115

Figure 4.3. Explanation of the method 119

Figure 4.4. Typical construction of an ANN with a single layer of input, hidden, and output. . 128

Figure 4.5. Visualization of the predicted, actual, and residual values. 131

Figure 4.6. Visualization of the predicted, actual, and residual values. 132

Figure 4.7. Visualization of the predicted, actual, and residual values. 132

Figure 4.8. Relationship of research efforts with the temporal surrogate models and connection between chapters of the dissertation. 133

Figure 5.1. Spatial prediction model. 153

Figure 5.2. Distribution of the crime over the Los Angeles census tracts. (produced by author) 156

Figure 5.3. Visualization of the patterns in each predictor/covariate over the LA Census Tracts. 158
Figure 5.4. Residual pattern of the linear Model 3. (Produced by author) 159
Figure 5.5. The AIC values for different bandwidths 161
Figure 5.6. Results of G, F and K statistics tests. 163
Figure 6.1. Triangle showing the relationships among three design criteria (Alizadeh et al., 2020a). 176
Figure 6.2. Hot rod rolling process (Alizadeh et al., 2019). 177
Figure 6.3. An example of a banded microstructure in 1020 steel consisting of ferrite (light) and pearlite (dark) (Jägle, 2007). 178
Figure 6.4. Comparison of polynomial response surface model predictions *(Nellippallil et al., 2018).* 180
Figure 6.5. Method for selecting the surrogate model based on time, size, and accuracy. 181
Figure 6.6. The experimental procedure. 182
Figure 6.7. RMSE in the prediction of surrogates in 10 runs. 188
Figure 6.8. RMS errors in the prediction of surrogates with different folds of data from 9 to 2.190
Figure 7.1. Relationship of research efforts with the temporal surrogate models and connection between chapters of the dissertation. 204
Figure 7.2. Three-layer structure of the surrogate model selection process. 220
Figure 7.3. AutoSM framework consists of two phases, the offline phase is used for the selector generation, and an online phase is used for the new data surrogate model type prediction 223
Figure 7.4. Distribution of the test data generated from theoretical problems 230
Figure 7.5. Hot rod rolling process 232
Figure 7.6. The geometry of elbow (a.), blowpipe (b.) and blowpipe system(c.) *(Cui et al., 2018).* 234
Figure 7.7. Experimental system for validating the accuracy and robustness of the selection model. 236
Figure 7.8. Predicted accuracy of hot rod rolling problem 241
Figure 7.9. Predicted accuracy of the blowpipe design problem 241
Figure 7.10. Frequency of best surrogate-optional types (Left figure) and predicted surrogate-optional types from AutoSM across 5 test problems (Right figure) 242
Figure 7.11. A statistical histogram of the relation between nonlinearity and accuracy. 243
Figure 7.12. Average of accuracy and robustness of each surrogate-optional type 246
Figure 7.13. Categorization of surrogate-optional types based on accuracy for multi-label CART 246

Figure 8.1. Total emissions in 2019 = 6,457 million metric tons of CO_2 equivalent (data source is (EPA, 2019) and figure is drawn by authors). 251

Figure 8.2. A network partitioned using the k-median algorithm..... 255

Figure 8.3. A two-echelon k-median network (left), and a two-echelon k-median and TSP network (right). 255

Figure 8.4. The proposed GSC network design model. 268

Figure 8.5. Municipalities of Puerto Rico (Wikiland, 2020) 274

Figure 8.6. Locations of customers (circles) and the stores (squares) that they connect to, in the emission minimizing system and operating cost minimizing systems. Distances are in kilometers (Williams et al., 2020). 274

Figure 8.7. San Juan operating cost, carbon tax, and emission minimizing scenarios. 279

Figure 8.8. Solution method..... 296

Figure 8.9. A two-echelon k-median network 299

Figure 8.10. Candidate stores and warehouse in Puerto Rico..... 301

Figure 8.11. Open stores and warehouse in Puerto Rico. 302

Figure 8.12. Candidate stores and warehouse on the east side of Puerto Rico. 303

Figure 8.13. Candidate stores and warehouse in the west side of Puerto Rico..... 304

Figure 8.14. Open stores and warehouse on the east side of Puerto Rico. 305

Figure 8.15. Open stores and warehouse on the west side of Puerto Rico. 306

Figure 8.16. Percentage of open stores with changing the number of warehouses in Puerto Rico. 306

Figure 8.17. Total percentage of open stores in Puerto Rico with one warehouse vs. two warehouses. 307

Figure 8.18. The value of deviation variables with W1=1/2 and other W=1/54. 309

Figure 8.19. Deviation plotted against iteration..... 311

Figure 8.20. Difference between the distance customer travels in in-person shopping and the distance delivery van travels in online shopping (the delivery tour). 314

Figure 8.21. Framework for multi-echelon, multi-commodity, multi-channel, supply chain design with climate change mitigation perspective. 315

Figure 8.22. Feasible solution for the bi-level problem..... 335

Figure 8.23. Tour distance in online shopping for each sample point and the corresponding cost of each tour. 336

Figure 8.24. Final tour for each decision (design) and mapping for the first sample (32 datapoints or designs). 337

Figure 8.25. Results of calculating covariance matrix of the 32 random solutions. 338

Figure 8.26. The module for calculating the μ and $\sigma 2$ and the likelihood function. 339

Figure 8.27. Results of calculating the likelihood function parameters. 340
Figure 8.28. Results of calculating $S2$(x). 341
Figure 8.29. Results for the bi-level model which is transferred to a single level model..... 341
Figure 8.30. Results for the lower level and upper-level objective function using the first set of sample solutions and the Kriging solution. 342
Figure 8.31. The lowest value of upper-level objective function among the first 33 solutions.. 343
Figure 8.32. The results for the second 32 samples for the tour finding (value of *f(x,y)*, the lower-level problem). 343
Figure 8.33.Value of f(x,y), the lower level prolem and F(x,y), the upper level prolem and the updated lowest solution for upper level (F) for the second 32 samples. 344
Figure 8.34. the hypothetical different between different percentages of online shoppers. 345
Figure 8.35. Difference scenarios based on different percentages of the online shoppers. 347
Figure 8.36. Total cost of the supply chain based on percentage of online shoppers. 347
Figure 9.1. The dissertation outline and the work accomplished during this PhD. 354
Figure 9.2. A visual description of the framework to provide guidance for researchers and practitioners to choose the most appropriate surrogate model. 355
Figure 9.3. A visual representation of the proposed approach for AutoSM selection in Chapter 7. 357
Figure 9.4. Research thrusts. 362
Figure 9.5. Components of a socio-cyber-physical system. 367
Figure 9.6. Linked socio-cyber-physical systems forming a production network. 368
Figure 9.7. Some social aspects of cyber-physical-social system. 370
Figure 9.8. Unexpected events affecting the supply chain. 370
Figure 9.9. Impact of unexpected events on SC and how digital thread monitors them. 371
Figure 9.10. How digital thread works. 371
Figure 9.11. The benefits of using digital threads in supply chains. 372
Figure 9.12. The evolution of digital threads in supply chains. 373
Figure 9.13. Future work based on my PhD. 374

Summary

Problem: Complex systems are composed of many interdependent subsystems with a level of complexity that exceeds the ability of a single designer. One way to address this problem is to partition the complex design problem into smaller, more manageable design tasks that can be handled by multiple design teams. Partitioning-based design methods are decision support tools that provide mathematical foundations, computational methods to create such design processes. Managing the interdependency among these subsystems is crucial and a successful design process should meet the requirements of the whole system which needs coordinating the solutions for all the partitions after all.

Approach: Partitioning and coordination should be performed to break down the system into subproblems, solve them and put these solutions together to come up with the ultimate system design. These two tasks of partitioning-coordinating are computationally demanding. Most of the proposed approaches are either computationally very expensive or applicable to only a narrow class of problems. These approaches also use exact methods and eliminate the uncertainty. To manage the computational complexity and uncertainty, we approximate each subproblem after partitioning the whole system. In engineering design, one way to approximate the reality is using surrogate models (SM) to replace the functions which are computationally expensive to solve. This task also is added to the proposed computational framework. Also, to automate the whole process, creating a knowledge-based reusable template for each of these three steps is required. Therefore, in this dissertation, we first partition/decompose the complex system, then, we approximate the subproblem of each partition. Afterwards, we apply coordination methods to guide the solutions of the partitions toward the ultimate integrated system design.

Validation: The partitioning-approximation-coordination design approach is validated using the validation square approach that consists of theoretical and empirical validation. Empirical validation of the design architecture is carried out using two industry driven problems namely the 'a hot rod rolling problem', 'a dam network design problem', 'a crime prediction problem' and 'a green supply chain design problem'. Specific sub-problems are formulated within these problem domains to address various research questions identified in this dissertation.

Contributions: The contributions from the dissertation are categorized into new knowledge in five research domains:

- Creating an approach to build ensemble of surrogate models when the data is limited – when the data is limited, replacing computationally expensive simulations with accurate, low-dimensional, and rapid surrogates is very important but non-trivial. Therefore, a cross-validation-based ensemble modeling approach is proposed.
- Using temporal and spatial analysis to manage the uncertainties - when the data is time-based (for example, in meteorological data analysis) and when we are dealing with geographical data (for example, in geographical information systems data analysis), instead of feature-based data analysis time series analysis and spatial statistics are required, respectively. Therefore, when the simulations are for time and space-based data, surrogate models need to be time and space-based. In surrogate modeling, there is a gap in time and space-based models which we address in this dissertation. We created, applied and evaluated the effectiveness of these models for a dam network planning and a crime prediction problem.
- Removing assumptions regarding the demand distributions in green supply chain networks – in the existent literature for supply chain network design, there is always assumptions about the

distribution of the demand. We remove this assumption in partition-approximate-compose of the green supply chain design problem.

- Creating new knowledge by proposing a coordination approach for a partitioned and approximated network design. A green supply chain under online (pull economy) and in-person (push economy) shopping channels is designed to demonstrate the utility of the proposed approach.

Besides the contributions I have made in this dissertation, I have identified new gaps and questions. The first gap in using data-driven approximation models in creating rule-based computational frameworks, knowledge-based platforms, and ontologies to manage the knowledge resulting from the modeling. I posed the question of "What is the mathematics underlying automating the data-driven modeling and approximation models?" to address this gap. The second gap is on embedding the social aspect in the cyber-physical-social systems. I posed the question of "social aspect of the cyber-physical-social systems". The third gap is on utilization of digital threads in green supply chain as a cyber-physical-social system. to address this gap, I pose the questions of: (i) how can seamless visibility across the supply chain enable informed decision making? (ii) how can prescriptive decisions enable supply chain planners to navigate through unforeseen and exceptional scenarios?

CHAPTER 1 MOTIVATION AND PROBLEM IDENTIFICATION

The foundational premise for this dissertation is that a solutions approach involving partitioning, approximation, and coordination to manage computational complexity. This framework is needed in complex engineered design systems, especially in cyber-physical-social systems. A cyber-physical-social systems design revolution is underway in the recent past where the focus is to build these systems in the context of a low-carbon economy and pull economy. Designing a cyber-physical-social system with plenty of interconnectivities is a complex problem. Adding the characteristics of a low carbon economy where online shopping is considered alongside the traditional in store shopping makes this design problem even more complex. Recent advancements in data gathering and analysis through machine learning and simulation-based approximations through surrogate modeling have made managing these complexities and associated uncertainties easier. Also, besides approximation, partitioning a cyber-physical-social system into manageable partitions is another way to manage the complexity. Also, this provides an opportunity to further use of approximation through surrogate modeling in portioned problems with less computational expense.

The major challenge arising in cyber-physical-social systems design is the management of uncertainty and complexity along with coordinating the solutions for the portioned system. Multi-level modeling approaches and approximation through surrogate models are usually domain-specific demanding considerable knowledge and insight in supply chains, transportations, healthcare and energy networks and thus corresponds to detailed design. System-based design approaches for cyber-physical-social systems are mostly domain-independent.

The conventional way of surrogate modeling only involves feature-based data analysis while a cyber-physical-social system design requires temporal, spatial, textual and imagery data analysis.

Also, conventional supply chain designs lacks considering pull economy's features including online shopping channel's dynamics with in-store shopping. Therefore, a multi-level partitioning and coordination approach to deal with multi-channel systems such a supply chain with in-store and online shopping is proposed in this dissertation.

In Chapter 1, a foundation is laid for achieving the goals addressed in this dissertation, where motivation, background and frame of reference in Sections 1.1 and 1.2, are presented which contains literature review and discussion on following topics: (1) approximation through surrogate modeling, (2) partitioning and coordination of multi-echelon, multi-channel and multi-commodity cyber-physical-social system, (3) pull-push economy under carbon mitigation scenarios, (4) challenges and research gaps in managing complexity and uncertainty in designing cyber-physical-social systems transitioning from push economy to pull economy in the context of a low carbon economy. Authors principal goal in this dissertation is identified by carrying out a gap analysis and hypothesis are laid to address these gaps. Research questions worthy of investigation are framed and the expected new knowledge on answering the research questions are identified. An overview of the hypothesis expected contributions and validation strategy are discussed in each section. The organization of the dissertation and a road map for accomplishing the chapters panned are presented in Chapter 1.

This chapter is revisited for checking structural soundness of the dissertation where literature review, design approach, developed method, and validation of hypotheses are discussed in following chapters.

1.1 Motivation for Managing Computational Complexity and Uncertainty in Designing Evolving CPS Systems

(Transitioning from Push to Pull, Low Carbon Economy Using Multi-Echelon Partitioning - Approximation - Coordination Approach)

Complex systems are composed of many interdependent subsystems with a level of complexity that exceeds the ability of an only one design team. We define complexity in the context of engineering design where the models are incomplete and inaccurate. Complex problems cannot be solved exactly and require approximating the reality. Also, they are complex due to the large number of system components which is required to be partitioned into manageable partitions. One way to address this problem is to partition the complex design problem into smaller, more manageable design tasks that can be handled by one single design team. Partitioning-based design methods are decision support tools that provide mathematical foundations, computational methods to create such design processes. Managing the interdependency among these subsystems is crucial and a successful design process should meet the requirements of the whole system which needs coordinating the solutions for all the partitions after all. Partitioning and coordination should be performed to implement the partitioning-based design: partitioning the system into subproblems and determining the coordination method to direct subproblem solutions toward the ultimate system design. These two tasks of partitioning-coordinating are computationally demanding. Most of the proposed approaches are either computationally very expensive or applicable to only a narrow class of problems. These approaches also use exact methods and eliminate the uncertainty. To manage the computational complexity and uncertainty, we approximate each subproblem after partitioning the whole system. In engineering design, one way to approximate the reality is using surrogate models (SM) to replace the functions which are computationally expensive to solve. This task also is added to the proposed computational framework. Also, to automate the whole process,

creating a knowledge-based reusable template for each of these three steps is required. Therefore, in this dissertation, I first partition/decompose the complex system, then, we approximate the subproblem of each partition. Afterwards, we apply coordination methods to guide the solutions of the partitions toward the ultimate system design.

1.1.1. Motivation of designing multi-echelon, multi-channel supply chain design in a low carbon, push-pull economy

A push economy is where suppliers work to secure products or inventory in anticipation of consumer demand and the push for the flow of the products is from the suppliers to the customers. One of the examples of a push economy is in-store shopping or bricks and mortar shopping channel. On the other hand, in a pull system, the supply chain only responds when there is consumer demand. Bricks and clicks channel where customers order online and pick up from the curb side as well as pure players which is pure online shopping are two of the examples of a pull economy (Fowler et al., 2019).

A low carbon economy is an economy based on sustainable actions, mainly focused on reducing or even sequestering the greenhouse gases (GHG) generated in the production chain, resulting in less environmental impact. To achieve a low-carbon economy, green supply chains are required. A green supply chain or sustainable network is defined as the operational management method to reduce the environmental impact along the life cycle of the green product, from the raw material to the end product and covers the entire life cycle of the product: manufacturing, storage, transport, marketing, use and disposal (Peng et al., 2020).

To study the dynamics of a low carbon, push-pull economy, a multi-echelon (including warehouses, stores and customers), multi-channel (including in-store and online shopping), a green supply chain is designed and modelled to calculate the difference in expected carbon emissions

between in-store shopping, curb side customer pickup versus e-commerce-based online retailing involving last mile delivery to customers' homes to quantify which channel has the least harmful impact on the environment.

1.1.2. Addressing the challenges of managing computational complexity and uncertainty in multi-echelon, multi-channel supply chain

(In a low carbon, pull economy)

There are some challenges associated with managing computational complexity and uncertainty in designing such a model and framework as the problem is multi-level and therefore computationally expensive. To deal with the computational expense, a clear boundary should be set around the problem. Therefore, several assumptions have been made:

Two level problem is studied including the in-store and online shopping channels, and a bi-level programming model is used to formulate this supply chain.

Also, greenness of the model is provided by monetizing the greenhouse gas emission due to transportation

Although several assumptions are made to manage the computational complexity, the problem is still an NP-Hard problem and cannot be solved exactly. Therefore, a surrogate-based approximation method is used to manage the computational complexity and uncertainty of the problem. To identify the appropriate approximation method, a critical evaluation of the literature and several contributions on surrogate models and approximation approaches are conducted in Chapters 2-7.

1.2 Gaps, Research Questions/Questions, Hypothesis, New Knowledge, Functionality, and Utility

Gap 1: We observe that a framework for selecting a more appropriate SM for a given function with specific requirements are lacking.

Research Question 1: To address Gap 1, we pose Research Question 1 in the form of three questions which answering them enables me to create New Knowledge 1: (1.1) What are the main classes of the design of experiment (DOE) methods, surrogate modeling methods and model-fitting methods based on the requirements of size, computational time, and accuracy? (1.2) Which surrogate modeling method is suitable based on the critical characteristics of the requirements of size, computational time, and accuracy? (1.3) Which DOE is suitable based on the critical characteristics of the requirements of size, computational time, and accuracy?

Hypothesis 1: Based on these three characteristics of time, size, and accuracy, we hypothesize that there are six different qualitative categories for the surrogate models through a critical evaluation of the literature.

New Knowledge 1: we come up with six detailed categories using them, we provide a framework for selecting an efficient surrogate modeling process to assist those who wish to select more appropriate surrogate modeling techniques for a given function. The value of such a framework is in providing practical guidance for researchers and practitioners in industry to choose the most appropriate surrogate model based on incomplete information about an engineering design problem. Another contribution is to use three drivers, namely, computational time, accuracy, and problem size instead of using a single measure that authors generally use in the published literature.

Functionality and Utility: Providing foundations for automating the SM selection process.

Gap 2: In engineering design, to replace costly computer simulations a single surrogate model is selected based on previous experience. There is a gap in SM pool of knowledge in creating **ensembles** (combinations) of SM when the data is **limited**.

Research Question 2: what is the mathematics to create an ensemble of SMs which is more accurate and less computationally expensive when the data is sparse and limited?

Hypothesis 2: we hypothesize, based on an analysis of the published literature, that fitting an ensemble of surrogates (EoS) based on cross-validation errors is more accurate but requires more computational time.

New Knowledge 2: A new method to build ensemble of SMs which is accurate and computationally inexpensive when the data is sparse.

Functionality and Utility: Enabled prediction with sparse data by ensemble of SM.

Gap 3: The choice of SM is extremely important since there are many types of SMs, and they also have different hyper-parameters. Traditional manual selection approaches are very computationally expensive, they lack interpretability and cannot be generalized.

Research Question 3: what is the mathematics to automate the SM selection?

Hypothesis 3: we hypothesize to build an interpretable decision tree to map four critical features, including problem scale, noise, size of sample and nonlinearity, to the types of SM and select the promising SM; then, use a genetic algorithm (GA) to find the appropriate hyper-parameters for each chosen SM.

New Knowledge 3: The new knowledge created through addressing this gap is a method to create a rule-based model for an automatic SM selection called AutoSM. The drastic increase in the selection pace by pre-screening of SM types based on selection rule extraction is the scientific

contribution of our proposed method. The proposed AutoSM, unlike previous EA-based automatic SM selection methods, is not a black box and is interpretable.

Functionality and Utility: SM selection easier and faster than existing selection methods by creating a systematic and organized automated selection process.

Gap 4: There is a gap in SM literature in dealing with time-based (temporal) data.

Research Question 4: What is the appropriate SM to use when the data includes time-dependent variables as predictors?

Hypothesis 4: we hypothesize by replacing the DOE with the time lags analysis, we find the SM which is useable in temporal variables. These SMs are better than classic time series analysis methods like ARIMA. Also, EoS are better than individual SM in temporal variables.

Key outcome 1: Dealing with temporal data by incorporating time series (lag analysis) with SM. This is new in surrogate modelling literature. Temporal data can be used in SM using different time lags. We also show that using EoS, we achieve more accurate predictions.

Functionality and Utility: Enabled dealing with temporal data in SM.

Gap 5: There is a gap in SM literature in dealing with space-based (spatial) data.

Research Question 5: What is the appropriate SM to use when the data includes space-dependent variables as predictors?

Hypothesis 5: we hypothesize by Replacing the design of experiments with the geographically weighted correlation analysis, we find the surrogate model which is useable with spatial variables.

Key outcome 2: Using the spatial statistics and particularly the concept of the geographically weighted regression in surrogate modelling, we created spatial SMs.

Functionality and Utility: Enabled dealing with spatial data in SM.

Gap 6: There is a gap in partitioning a complex design problem, approximate and coordinate the solutions of the partitions.

Research Question 6: What is the mathematics to partition and coordinate a multi-channel, multi-echelon supply chain design problem?

Hypothesis 6: we hypothesize the research question can be addressed by designing a multi-channel, multi-echelon supply chain in the form of a bi-level model where the upper level is to design the supply chain, including identifying the layout, the number of warehouses and stores, while the lower level is to identify the tour in online shopping. Then, using approximation (surrogate modelling) in the partitioning-coordination framework and making it partitioning-approximation-coordination framework, we propose an approach using surrogate approximation-based model where we manage the computational complexity by iteratively approximating the delivery van tour function in online shopping (which is also called the lower-level function) in the multi-channel, multi-echelon green supply chain.

New Knowledge 4: Creating an algorithm to solve a complex problem (an NP hard bi-level programming problem) using partitioning, approximation, and coordination.

Functionality and Utility: A method for calculating the difference in expected carbon emissions between in-store shopping, curb side customer pickup versus e-commerce-based online retailing involving last mile delivery to customers' homes to quantify which channel has the least harmful impact on the environment. SM approximations are created during the generations of an evolutionary algorithm, where the population members serve as the samples for creating the approximations. One of the important features of the proposed algorithm is the creation of an

auxiliary problem using the Kriging-based metamodel of the lower-level optimal value function that solves an approximate relaxation of the bilevel problem. Using the auxiliary problem when used for local search we are able to accelerate the tour finding and supply chain design toward the bilevel solution.

1.3 Validation Square

1.3.1. Theoretical structural validity – Is the design method internally consistent?

The proposed design method is based on partitioning-based design method. The base method was implemented in previous example problems. The internal consistency of the method is supported by its logical process and successful implementation. The internal consistency of all applications on test problems are based on agreement with existing design theory and effectiveness of implementation in an interdependent complex system architecture.

1.3.2. Empirical structural validity – Are the example problems used in modeling the method appropriate choices?

The example problems studied in this dissertation are complex systems with interdependent network architecture. The dam network planning problem is a network with interdependencies among dams, users and fish which should be partitioned, approximated and composed to obtain the ultimate system result. The crime forecasting problem is a network problem with geographic and demographic interdependencies. The green supply chain design problem is a network problem with interdependencies among different echelons, including supplier, customer, retailor and the warehouse besides different channels, including online and in-person shopping channels. The hot rod rolling problem is also a multi-stage problem with vertical and horizontal interdependencies.

1.3.3. Empirical performance validity – Does the application of the method to the problems in the dissertation produce practical results?

Results from the example problems demonstrate the benefits of using approximating the reality in partitioning/coordination framework over traditional design methods which do not consider the uncertainty management.

1.3.4. Theoretical performance validity – Does the application of the method to other problems produce practical results?

The previous three steps in the Validation Square build confidence in the expansion of the proposed approach to other similar example problems. Based on the internal consistency of the proposed approach, the appropriateness of the selected example problems, and the effective implementation of the proposed approach in solving the example problems, one can one can conclude that applying the proposed approach to similar problems are produce practical and desirable results.

1.4 Contributions

The contributions from the dissertation are categorized into new knowledge in five research domains:

- Creating an approach to build ensemble of surrogate models when the data is limited – when the data is limited, replacing computationally expensive simulations with accurate, low-dimensional, and rapid surrogates is very important but non-trivial. Therefore, a cross-validation-based ensemble modeling approach is proposed (Chapter 6).
- Creating practical guide for researchers and practitioners in industry to choose the most appropriate surrogate model based on incomplete information about an engineering design problem (Chapters 2 and 3).

- Creating a new method to build ensemble of SMs which is accurate and computationally inexpensive when the data is sparse (Chapter 6).
- Using temporal and spatial analysis to manage the uncertainties - when the data is time-based (for example, in meteorological data analysis) and when we are dealing with geographical data (for example, in geographical information systems data analysis), instead of feature-based data analysis time series analysis and spatial statistics are required, respectively. Therefore, when the simulations are for time and space-based data, surrogate models need to be time and space-based. In surrogate modeling, there is a gap in time and space-based models which we address in this dissertation. We created, applied, and evaluated the effectiveness of these models for a dam network planning and a crime prediction problem (Chapters 4 and 5).
- Removing assumptions regarding the demand distributions in green supply chain networks – in the existent literature for supply chain network design, there is always assumptions about the distribution of the demand. We remove this assumption in partition-approximate-compose of the green supply chain design problem (Chapter 8).
- Creating knowledge-based reusable templates for each of the three steps – we create reusable templates for each of three steps of partition-approximate-compose approach in knowledge-based platform which is available in the cloud. These reusable templates enable the designers and users with low domain knowledge to use these approaches in an automatic way (Chapter 7).
- Creating an algorithm to solve a complex problem (an NP hard bi-level programming problem) using partitioning, approximation, and coordination (Chapter 8).

1.5 Research Thrusts and Layout of the Dissertation

In , different research thrusts that are completed in this dissertation and how they are connected are shown. Also, the connection between these research projects and their correspondent contributions as well as their connection to each chapter of the dissertation are shown in Figure 1.1 and Figure 1.3. The connection between chapters and research questions, hypothesis and how we verify and validate the structure of my dissertation is shown in Figure 1.3. Based on the proposed framework in and the skeleton of the dissertation in Figure 1.2, the layout of the dissertation is created and shown in Figure 1.4.

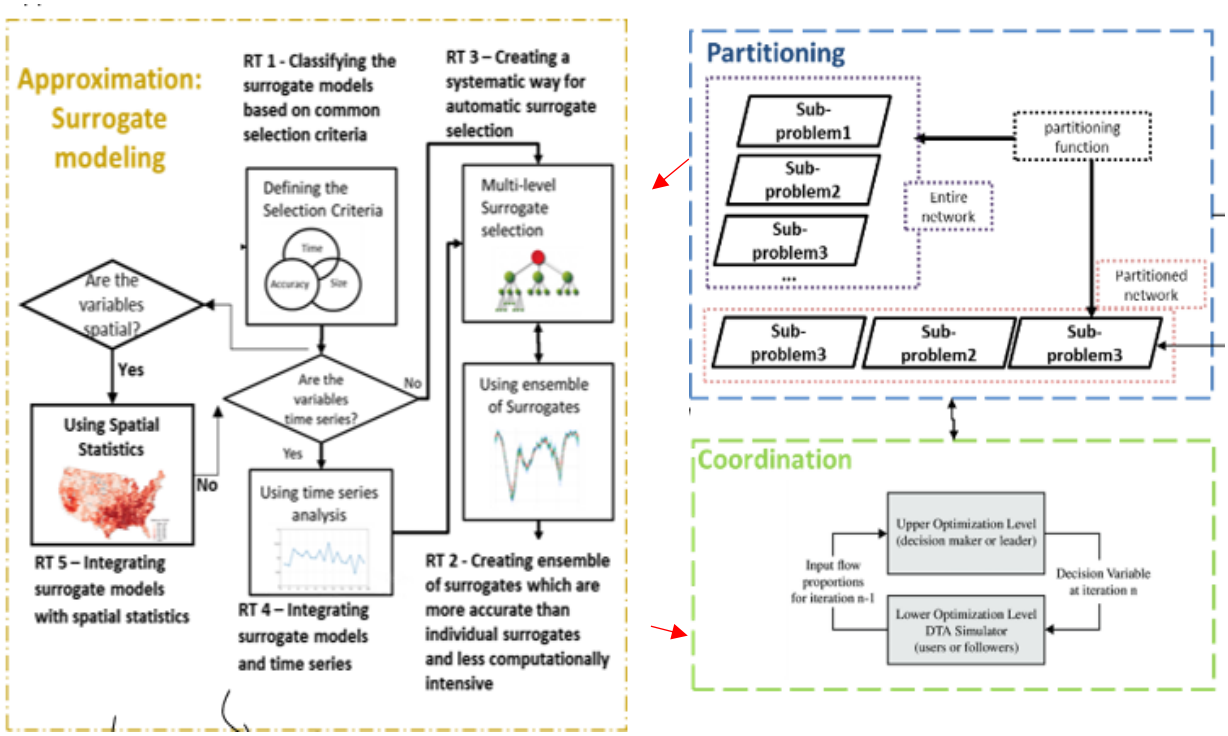


Figure 1.1. Partitioning-approximation-coordination framework.

As shown in Figure 1.2, four research questions are posed to address the gaps in the dissertation. Gap 1, which is on classifying the surrogate models based on common selection criteria is addressed in Chapters 2 and 3. Also, Gap 2 on creating EoS which are more accurate than individual surrogates is addressed in Chapter 6.

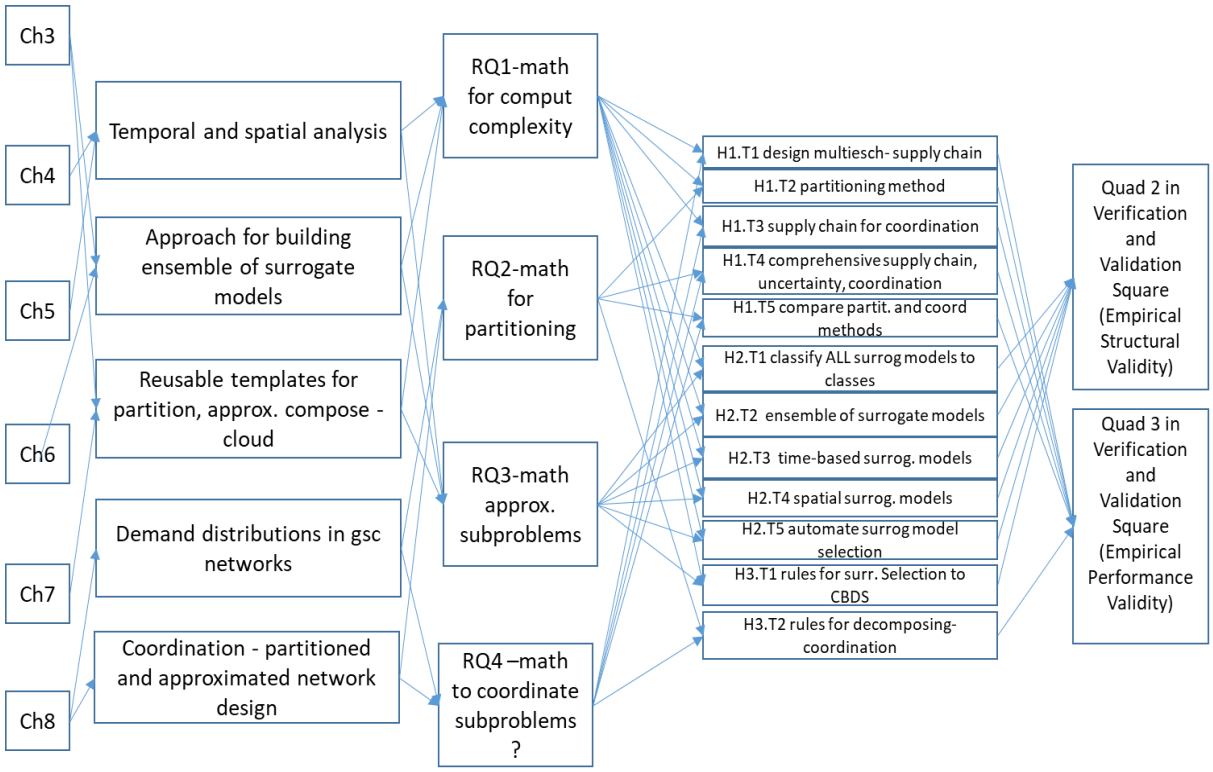


Figure 1.2. Skeleton of the dissertation.

Gap 3 on creating a systematic way for model selection is addressed in Chapter 7 while Gaps 4 and 5 on temporal and spatial surrogate models are addressed in Chapters 4 and 5, respectively. Finally, Gap 8, which is on creating the partitioning, approximation and coordination is addressed in Chapter 8.

1.6 Verification and Validation of Chapters of Dissertation

In Figure 1.4, we show how we verify and validate the organization and structure of my dissertation chapters using the Verification and Validation Square (Seepersad et al., 2006). Please note that this is only to verify and validate the structure and organization of the dissertation and we verify and validate the mathematics and knowledge that we create through my dissertation within in each chapter using the proper mathematics, formulation, modeling, and methods.

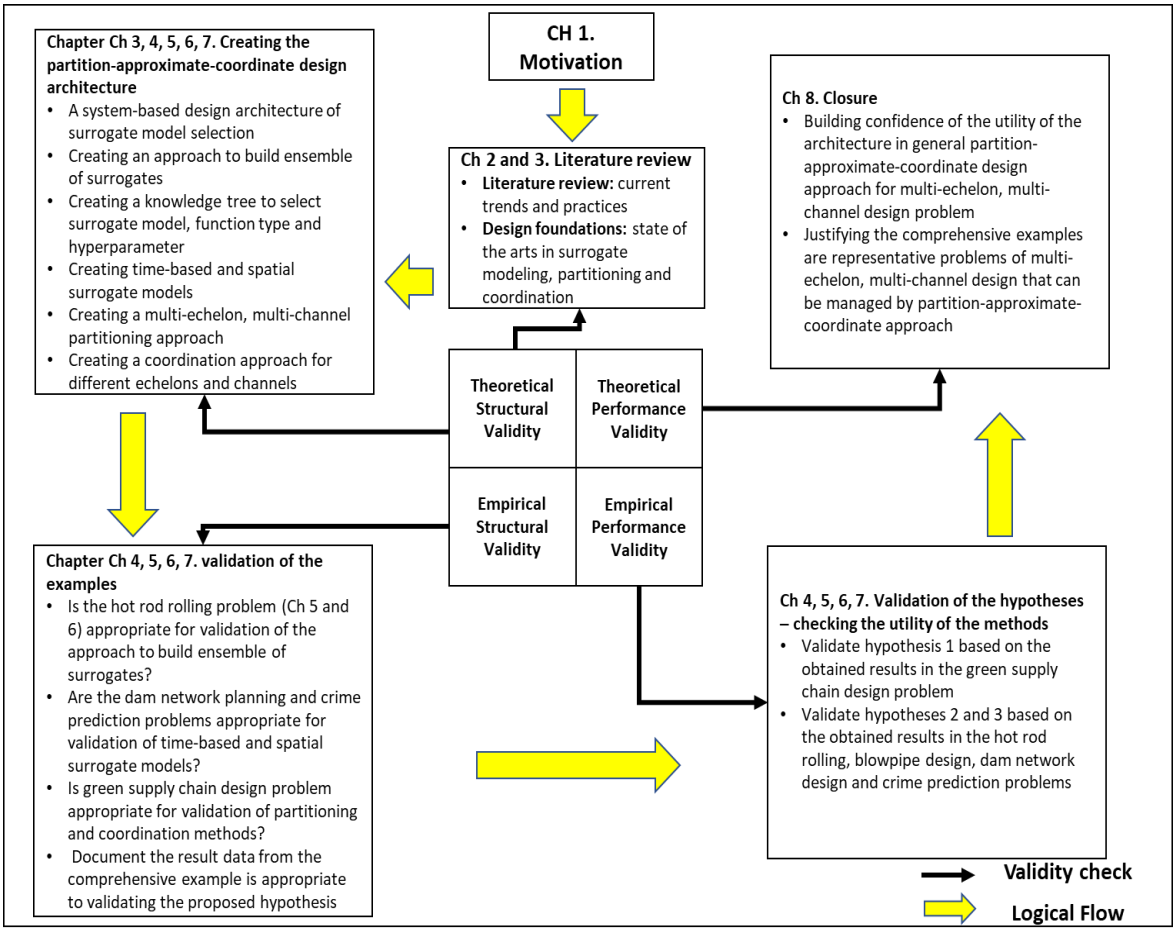


Figure 1.3. Organization of Dissertation Chapters according to Verification and Validation Square.

Table 1.1. Verification types and their associated chapters

Validation type	**Chapter**
Theoretical structural validity – is the partition-approximate-coordinate design architecture internally consistent?	Chapters 1, 2, 3, 4, 5, 6, 7
Empirical structural validity – Are the example problems appropriately chosen?	Chapters 4, 5, 6, 7
Empirical Performance Validity – Does the application of method to the sample problems produce practical results?	Chapters 4, 5, 6, 7
Theoretical Performance Validity – Is the partition-approximate-coordinate design architecture applicable for the other problems?	Chapter 7

Table 1.2. Connection between research questions, chapters, and validation square

Research Questions (RQ)	1	2	3	4	5	6	7	8
RQ1: What are the main classes of SMs based on the requirements of size, computational time, and accuracy?	**TSV**	**TSV**	**TSV**	**TSV** **EPV**	**EPV**	**EPV**	**SEV** **EPV**	**TPV**
RQ2: what is the mathematics to create an ensemble of SMs which are more accurate and less computationally expensive when the data is sparse and limited?	**TSV**	**TSV**	**TSV**				**ESV** **EPV**	**TPV**
RQ3: what is the mathematics to automate the SM selection?	**TSV**	**TSV**	**TSV** **ESV**	**TSV** **ESV**	**ESV** **EPV**	**ESV** **EPV**	**EPV**	**TPV**
RQ4: What is the appropriate SM to use when the data includes time-dependent variables as predictors?	**TSV**	**TSV**	**TSV**				**ESV** **EPV**	**TPV**
RQ5: What is the appropriate SM to use when the data includes space-dependent variables as predictors?	**TSV**	**TSV**	**TSV**				**ESV** **EPV**	**TPV**
RQ6: What is the mathematics to partition and coordinate a multi-channel, multi-echelon supply chain design problem?	**TSV**	**TSV**	**TSV**				**ESV** **EPV**	**TPV**

1.7 Dissertation Overview and Roadmap

The relationship of research efforts with the constructs of the design architecture developed is shown in Figure 1.2. An overview of this dissertation is presented as roadmap in Figure 1.4. The figure is intended to help navigate through the dissertation and develop an overall picture as to what is discussed in each chapter thereby establish context.

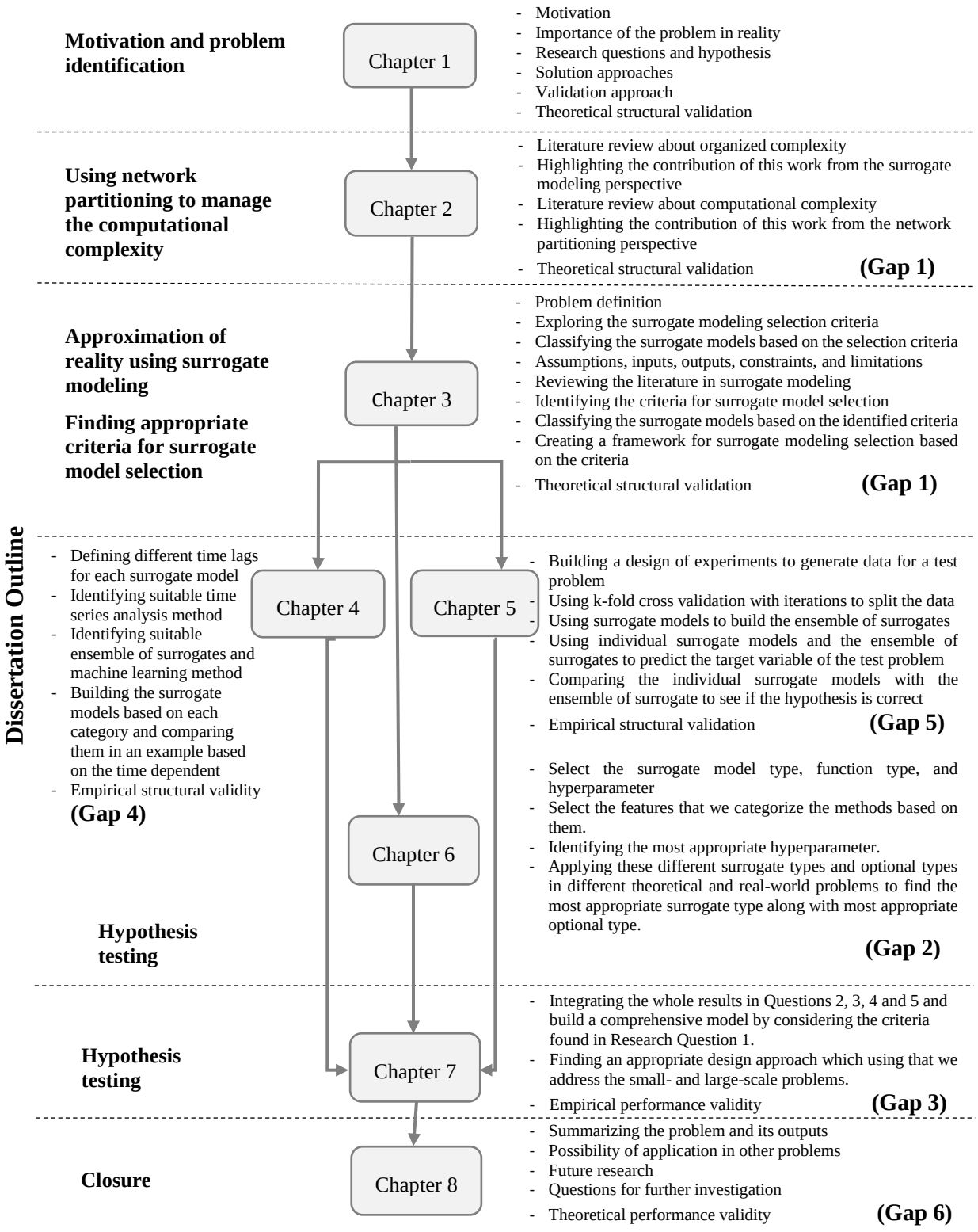


Figure 1.4. The dissertation overview and roadmap.

Now, that the outline, layout, and overall theme of the dissertation is established, we provide the critical evaluation of the literature on surrogate modelling as the first portion of this dissertation on approximation methods to replace computationally expensive models and manage complexity and uncertainty in engineering design problems. Also, we provide the frame of reference on complexity definition and managing computational complexity and uncertainty in designing multi-echelon, multi-channel, multi-commodity systems design for evolving cyber-physical-social systems in a low-carbon pull economy.

CHAPTER 2 FRAME OF REFENCE AND LITERATURE REVIEW

Managing Computational Complexity and Uncertainty in Designing Evolving Cyber-Physical-Social Systems Transitioning from Push to Pull, Low Carbon Economy Using Multi-Echelon Partitioning - Approximation - Coordination Approach – Current Trends and Practices

In Chapter 2, the frame of reference on the efforts associated with partitioning and coordination is given. Several critical issues associated with the current capabilities of partitioning and coordination is discussed in this chapter. Some of the major elements of computational complexity, multi-echelon and multi-channel green supply chain design are discussed in this chapter. The relationship of these research efforts reviewed in this chapter with the constructs of the systematic approach developed in this dissertation is highlighted in Figure 2.1.

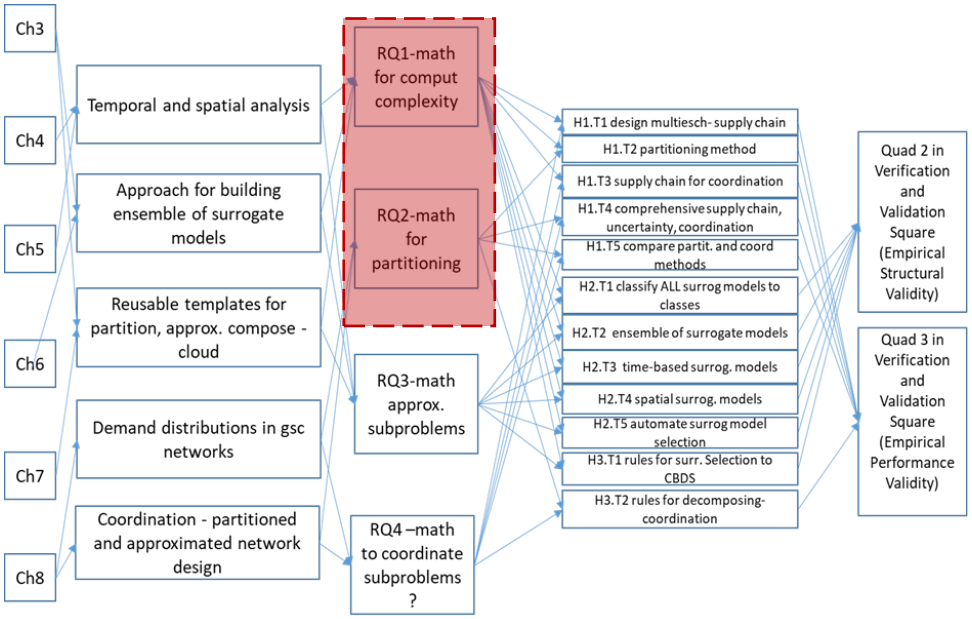


Figure 2.1. Relationship of research efforts with the constructs of the systems-based design architecture and connection between chapters of the dissertation.

In Section 1.1, the importance of managing computational complexity and uncertainty in designing multi-echelon, multi-channel supply chain is explained. In Section 2.2, the necessity of

creating a partitioning-approximation-coordination design approach is discussed. Also, in Section 2.3, the structure of the partitioning-based decision-making approach is articulated. Then, in Section 2.4, approximation of reality using surrogate modeling is elaborated in a way that they can be used in managing computational complexity. In Section 2.5, the use of network partitioning and coordination in managing computational complexity in a green supply chain design is explained.

2.1 Managing Computational Complexity and Uncertainty in Designing Multi-Echelon, Multi-Channel Supply Chain

(For Evolving Cyber-Physical-Social Systems in A Low-Carbon Push-Pull Economy)

Climate change is caused by carbon emissions and 70% of carbon emissions are generated within supply chains. So, designing green supply chains (GSC) is critical in mitigating climate change. Also, online shopping has gained huge share of retail sales during the past years, especially during the pandemic. For example, Walmart year over year growth in online sales reached to 97%. This provides a huge opportunity to design greener supply chains and mitigate climate change. The significance of this work is in enabling retailers like Walmart to:

- adapt post Covid-19 market and build greener supply chains
- design a new business model transitioning from a push economy to an on-line based pull economy.
- use predictive analytics to simulate the demand
- show whether traditional or online shopping is greener
- design a green and low-cost configuration.

The mathematics we created in this chapter is not specific for on-line retailing and can be used in designing any evolving cyber-physical-social systems such as healthcare, manufacturing, energy, and transportation.

2.1.1. Motivation of designing multi-echelon, multi-channel supply chain design in a low carbon, push-pull economy

A push economy is where suppliers work to secure products or inventory in anticipation of consumer demand and the push for the flow of the products is from the suppliers to the customers. One of the examples of a push economy is in-store shopping or bricks and mortar shopping channel. On the other hand, in a pull system, the supply chain only responds when there is consumer demand. Bricks and clicks in shopping where customers order online and pick up from the curb side as well as pure players which is pure online shopping are two of the examples of a pull economy (Fowler et al., 2019).

A low carbon economy is an economy based on sustainable actions, mainly focused on reducing or even sequestering the greenhouse gases (GHG) generated in the production chain, resulting in less environmental impact. To achieve a low-carbon economy, green supply chains are required. A green supply chain or sustainable network is defined as the operational management method to reduce the environmental impact along the life cycle of the green product, from the raw material to the end product and covers the entire life cycle of the product: manufacturing, storage, transport, marketing, use and disposal (Peng et al., 2020).

To study the dynamics of a low carbon, push-pull economy, a multi-echelon (including warehouses, stores and customers), multi-channel (including in-store and online shopping), green supply chain is designed and modelled to calculate the difference in expected carbon emissions between in-store shopping, curb side customer pickup versus e-commerce-based online retailing involving last mile delivery to customers' homes to quantify which channel has the least harmful impact on the environment.

2.1.2. Addressing the challenges of managing computational complexity and uncertainty in designing multi-echelon, multi-channel supply chain

(in a low carbon, push-pull economy)

In this dissertation, we define computational complexity in the context of engineering design where the models are incomplete and inaccurate. Computationally complex problems cannot be solved exactly and require approximating the reality. Also, they are computationally complex due to the large number of system components which is required to be partitioned into manageable partitions. There are some challenges associated with managing computational complexity and uncertainty in designing such a model and framework as the problem is multi-level and therefore computationally expensive. To deal with the computational expense, a clear boundary should be set around the problem. Therefore, several assumptions have been made: (i) Two level problem is studied including the in-store and online shopping channels, and a bi-level programming model is used to model this supply chain; (II) Greenness of the model is provided by monetizing the greenhouse gas emission.

Although several assumptions are made to manage the computational complexity, the problem is still an NP-Hard problem and cannot be solved exactly. Therefore, a surrogate-based approximation method is used to manage the computational complexity and uncertainty of the problem. To identify the appropriate approximation method, a critical evaluation of the literature and several contributions on surrogate models and approximation approaches is conducted in Chapters 2-7.

2.2 Creating Partitioning-Approximation-Coordination Design Approach

To create a comprehensive problem to cover all three portions of this dissertation, partitioning, approximation, and coordination, we use a multi-channel (online and in-store shopping), multi-

echelon (warehouse, store, and customer), green supply chain problem. To model and solve this problem, partitioning is required to partition the supply chain into smaller partitions and assign a store for each partition. Also, because there are two channels, including online and in store, a bi-level programming is needed where the lower level is to model the tour distance finding problem in online shopping and upper level is to model the supply chain design problem where the layout, number and size of the stores and warehouses are determined. This bi-level problem is NP-hard and nontrivial, so, we need approximation to solve it. Finally, since in the bi-level problem, upper level and lower-level problems must be partitioned and approximated concurrently, we need to coordinate these problems.

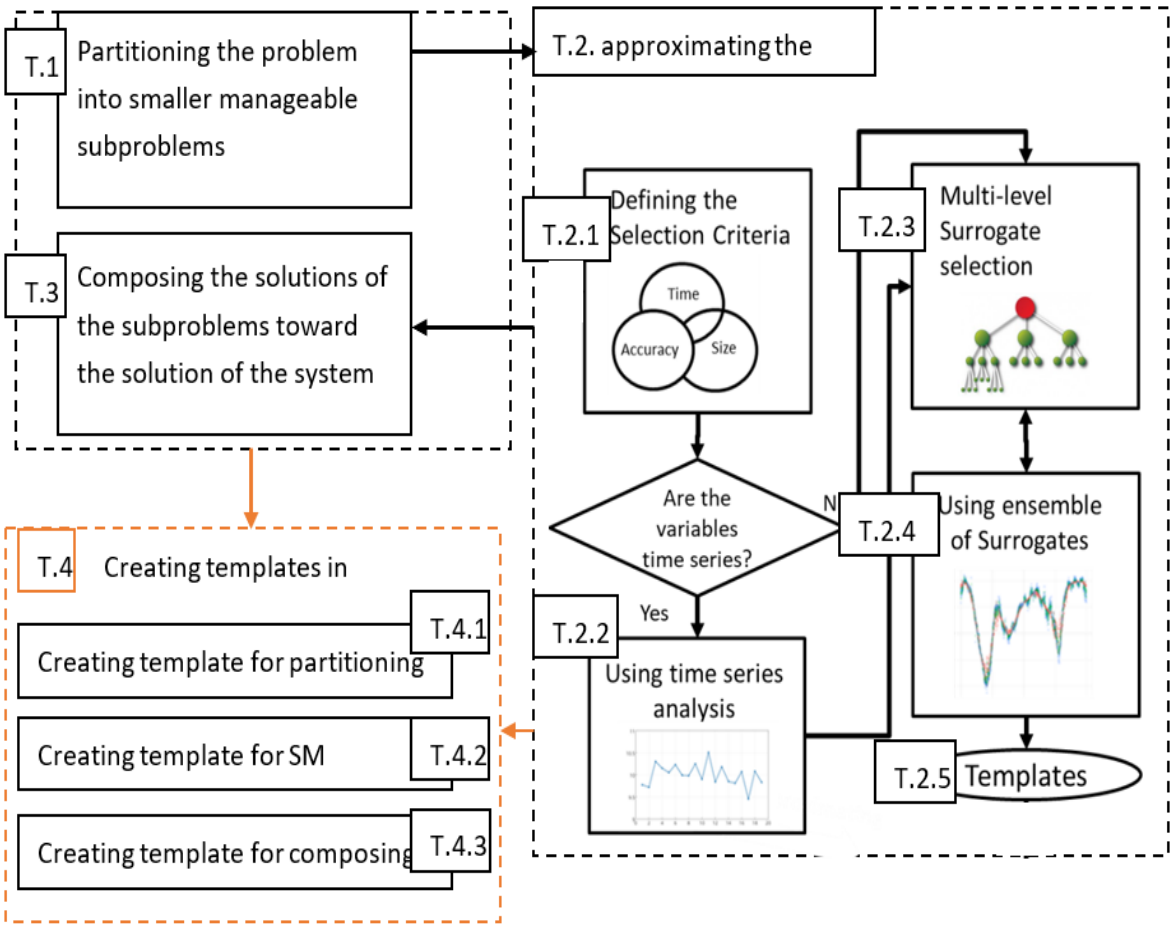


Figure 2.2. The idea of partitioning and coordination.

2.2.1. Problem statement of managing computational complexity using partitioning and coordination

As shown in Figure 2.2, for managing computational complexity which is the topic of this dissertation, complex systems are partitioned into interdependent subsystems which are more manageable design tasks that can be handled by one single design team (T.1 in Figure 2.2). Managing the interdependency among these subsystems is crucial and a successful design process should meet the requirements of the whole system which needs composing the solutions for all the partitions after all.

Therefore, two tasks should be performed to implement the partitioning-based design: (T.1) partitioning the system into subproblems and (T.3) determining the composing method to direct subproblem solutions toward the ultimate system design. These two tasks of partitioning-composing are computationally demanding. Most of the proposed approaches are either computationally very expensive or applicable to only a narrow class of problems. These approaches also use exact methods and eliminate the uncertainty.

To manage the computational complexity and uncertainty, we approximate each subproblem after partitioning the whole system. In engineering design, one way to approximate the reality is using surrogate models (SM) to replace the functions which are computationally expensive to solve (T.2 in Figure 2.2). This task also is added to the proposed computational framework. Also, to automate the whole process, creating a knowledge-based reusable template for each of these three steps is required (T.4). Therefore, in this dissertation, we first partition/decompose the complex system, then, we approximate the subproblem of each partition. Afterwards, we apply composing methods to guide the solutions of the partitions toward the ultimate system design.

As shown in Figure 2.2, managing computational complexity which is the topic of my dissertation starts with the question of "is the problem huge?" If the problem is huge, it means we have computational complexity and we need to partition the problem. Partitioning-based methods are appropriate when systems are large and sparsely connected (Sobieszczanski-Sobieski and Haftka, 1997), when the design environment is distributed (Chen et al., 2000), or when specialized computational algorithms can be exploited for solving particular subproblems (Kusiak and Larson, 1995). The techniques introduced in this article can be used to assess quantitively whether partitioning-based computational models are appropriate for a particular problem.

2.2.2. Using network partitioning to manage the computational complexity

When the size of a network is huge, partitioning methods are used to deal with the computational complexity. Based on the gaps and limitations of the methods in the literature, there are four main classes forming partitioning approaches: (i) Graph partitioning, (ii) Hierarchical clustering, (iii) Partitional clustering, (iv) Spectral clustering; from which the spectral clustering is the only approach that is used for both partitioning of a network and then composition of the partitioned network. So, we use that.

2.3 Partitioning-Based Decision-Making Approach (T1 in Figure 2.2)

A typical system design approach used in industry involves designing subsystems in sequence. For example, in design-based design where the decision is in different horizontal and vertical levels, this approach does not account explicitly for all design interactions. A simplified schematic of this process of decision-based design process where each decision in each level interacts with other decisions is shown in Figure 2.3.

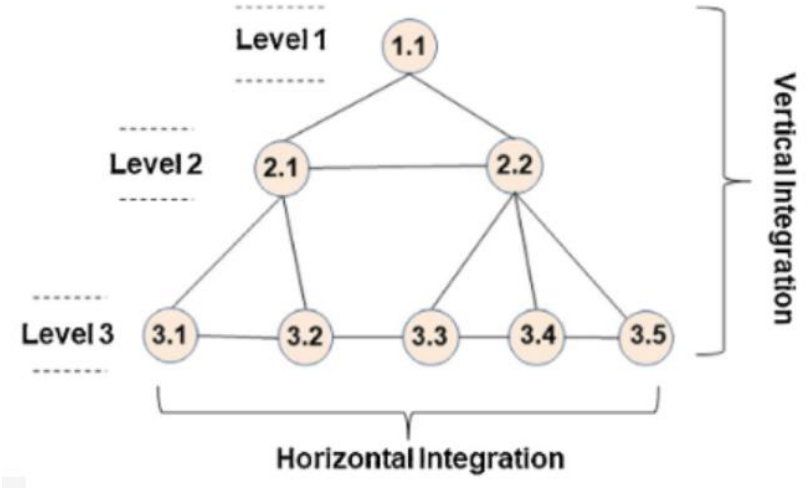
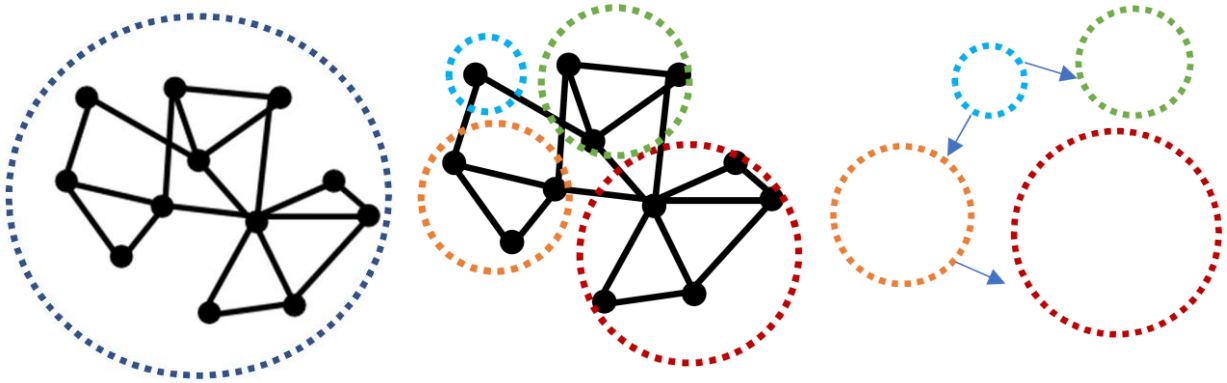


Figure 2.3. A partition hierarchy of a complex system.

In this example, the decision-based design (DBD) is performed first and is based on horizontal and vertical-level requirements. DBD also depends on the needs of the other subsystems, and they can be designed simultaneously. Similar to Figure 2.3, a lot of design problems do not have consecutive processes and they follow a network architecture. In such cases a network-based approach is needed. In network-based approach to reduce the computational complexity different approaches are taken. One of the strongest methods are used for this purpose are partitioning-based decision-making approaches.

Generally, partitioning-based approaches developed for situations where several engineering analyses must be integrated for designing a single component or product, where each analysis relates to a different aspect of the same component. Applying the decomposition-based decision-making needs the completion of two important preliminary steps. First, a system partition must be defined, and then a strategy for coordinating the solution of the resulting subproblems must be constructed (T1 in Figure 2.1). Making partitioning and coordination decisions can be viewed as a preprocessing step for appropriate system design. Figure 2.3 illustrates these preliminary steps. The original, unpartitioned system is depicted in Figure 2.4.a., where the vertices represent

components of a system or analyses pertinent to system design, and the edges connecting the vertices represent interactions between the components or analyses. The first step is to decide which subproblem each component should belong to. The outcome of this step is a system partition, shown in Figure 2.4.b. Once the partition is defined, a coordination strategy can be devised (T2). An important aspect of many coordination strategies is the subproblem solution sequence, illustrated in Figure 2.4.c.



a. Unpartitioned system

b. system partitioned into sub-problems

c. sub-problem coordination strategy

Figure 2.4. Process for implementing partitioning-based design.

The way a network is partitioned influence what interaction patterns are most effective. In addition, the type of interaction patterns desired (e.g., hierarchical vs. nonhierarchical) influence partitioning decisions. These two sets of decisions are coupled. Moving back to the context of petitioning-based design, system partitioning, and coordination decisions are also coupled. How a system is partitioned influence coordination decisions, and vice versa. This relationship is pictured in Figure. 2.5. Proving and investigating the relationship between partitioning and coordination decisions is the primary theme of this dissertation. Partitioning and coordination decisions have been studied independently, but the relationship between them has not been systematically analyzed. Subsequent chapters show that partitioning and coordination decisions are in fact coupled. Techniques are introduced for analyzing these decisions, studying intrinsic tradeoffs, and

making partitioning and coordination decisions for important classes of partitioning-based design methods.

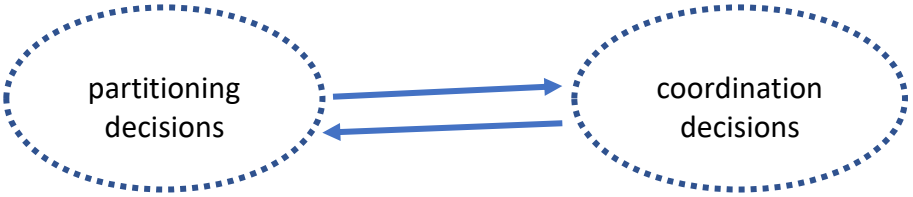


Figure 2.5. Finding coupling approaches between partitioning and coordination decisions

2.4 Approximation of Reality Using Surrogate Modeling (T2 in Figure 2.1)

In model-based realization of complex systems, we are forced to address the issue of computational complexity. One critical issue that must be addressed is the approximation of reality using surrogate models to replace expensive simulation models of engineering problems. So, approximating the problem decreases the computational cost of the method dramatically. However, a trade-off among the accuracy, dimension of the problem and computational time is required. Therefore, we are looking at different problems with different approaches. We hypothesize that a mathematical framework is a solution for the trade-off among accuracy, dimension of the problem and computational time.

2.4.1. Finding appropriate criteria for surrogate model selection (T2.1 in Figure 2.1)

To find the appropriate set of criteria, we need to critically evaluate the literature. For selecting appropriate surrogate modeling methods for a given function with specific requirements there is a gap for a selection framework based on appropriate criteria. To address this gap, we first need to find the appropriate criteria as the main criteria for automated surrogate selection. Second, we need to find the categories of design of experiments, surrogate modeling and model fitting methods based on these criteria.

2.4.2. Using time dependent features in surrogate modeling (T2.2 in Figure 2.1)

We substitute expensive experimentation and simulation with efficient surrogate models to address this computational complexity. Sometimes, the challenge is that the target variable which we want to predict is time dependent and we must consider the effect of the historical data on the target variable. In surrogate modeling literature, the is a gap of dealing with time dependent variables. In general, when the variable is time dependent, each time lag of the variable is considered as a new predictor and is taken as a non-time dependent variable into account. However, this method is ineffective when the number of time dependent variables is increased. Many possible scenarios are neglected in this method which decrease the reliability of the solution. Therefore, to address the gap of time dependent surrogate models, we integrate time series analysis and machine learning with surrogate models to predict the target variable. To examine the impact of time series in surrogate models, we use different time lags as features within the surrogate model and predict the target variable for different time steps ahead.

2.4.3. Using machine learning to automatically find appropriate function types and hyper-parameters

For effective application of surrogate models, the choice of surrogate models is extremely important. However, existing approaches often lack considering different function types and hyper-parameter values for the surrogate models. To address this gap, we need to create new mathematics to automatically identify the appropriate function type for each surrogate model and identify the appropriate value for the hyper-parameters within the function types to explore different prediction scenarios considering different predictors. We particularly need decision trees as interpretable machine learning method to automatically adjust the appropriate predictors. We incorporate the consideration of the features of the problem and demands form designers; and then we use it to develop an interpretable model to reduce the number of surrogate model candidates,

which greatly improves the model selection speed. We also illustrate a tree structure with summarized model selection rules which also provides suggestions for choosing fast real-time/run-time/online model selections.

2.4.4. Using ensemble of surrogates to use the advantages of various individual surrogates when the sample is undersized (T2.4 in Figure 2.1)

Surrogate models are often used instead of costly simulations in engineering problems. In general, an individual surrogate is selected according to the knowledge based on previous experience. Studies show that fitting an ensemble of surrogates based on cross-validation errors is more accurate but yields a more time-consuming approximation. There is a gap of using ensemble of surrogates based on a weighted average surrogate of individual surrogates based on overall error when the sample data is undersized. This kind of ensemble of surrogates have higher accuracy than individual surrogates even when fewer data points are used and so it demands less computational time. Also, the use of ensemble of surrogates results in accurate and relatively insensitive predictions at a reduced computational expense. This is new because it is used for a problem which ensemble of surrogates has not been used for it before.

2.4.5. Automating the process of surrogate modeling (T4 in Figure 2.1)

In simulation-based realization of complex systems, there is a gap of a dynamic and fully automated framework for surrogate model selection. In this dissertation, we want to fill this gap of existing literature and need of academic and practical world by considering appropriate set of surrogate models and the appropriate set of criteria to choose the surrogate models.

2.5 Using Network Partitioning to Design a Green Supply Chain

In supply chain network design for a manufacturing or material design process, a retailer may determine the number and locations of facilities based on the cost of opening the facility, the cost

of a customer driving to the facility, and the cost of a replenishment truck driving to the facility from a warehouse. However, this does not include the system's greenhouse gas (GHG) emissions. Given the existential threat posed by global warming, it is pertinent to consider how the design of this system affects its GHG emissions. In this section, we consider a green supply chain (GSC), in which GHG is minimized. We model the supply chain as a network of customers and store locations, with customers driving in cars to and from stores and the retailer resupplying the stores from a central warehouse. The number and location of stores is determined while designing the supply chain.

We consider the GSC problem to be an instance of the general k-median problem. Our contributions are (1) to remove the assumption of uniform demand, and instead build a model of a GSC based on real-world population data; (2) to model the GSC as a two-echelon k-median problem. For input, we used a high-precision population density map of Puerto Rico, which provides population density at a resolution of 30 meters by 30 meters. Data is stored as a 3-tuple of the x-y coordinates of each location and the number of people living at that location [116]. We analyzed two subsets of this data. The first contains the entire island, which has an area of 9,104 km^2 (170 km by 60 km), and a population of 3,994,259 people. The second contains the city of San Juan, Puerto Rico's largest city and capital. We considered both the city and the surrounding area, sub-setting a region with an area of roughly 2,500 km and a population of 2,199,923.

We consider two extreme scenarios: one that minimizes only the retailer's operating cost, and one that minimizes only emissions. After designing the GSC, we conduct a sensitivity analysis and consider the effect of a carbon tax in encouraging a greener system. We also consider various scenarios under which emissions might increase or decrease. Based on our results, specific scenarios can lead to a lower overall GHG emission. For example, doubling the fuel efficiency of

cars decreases emissions by 46% compared to the baseline scenario. The proposed design approach is not limited to GSC design and can be extended to other design problems, including selecting material suppliers, locating manufacturing facilities, and healthcare provider networks.

2.6 Role of Chapter 2 in This Dissertation

In Chapter 2, we provided the frame of reference on surrogate modeling as the first portion of my dissertation on approximation methods to replace computationally expensive models and manage complexity and uncertainty in engineering design problems. Also, we provided the frame of reference on complexity definition and managing computational complexity and uncertainty in designing multi-echelon, multi-channel, multi-commodity systems design for evolving cyber-physical-social systems in a low-carbon pull economy. Now, that the frame of reference on surrogate modeling as the approximation methods for managing computational complexity in engineering design problems as well as the frame of reference for partition and coordination of complex design problems are provided, we critically evaluate the literature on surrogate modeling in Chapter 3.

CHAPTER 3 SURROGATE MODEL CLASSIFICATION AND SELECTION CRITERIA

In Chapter 3, a review of the existing efforts associated with surrogate modelling and partitioning, and coordination is given. Several critical issues associated with the current capabilities of surrogate modelling is discussed in this chapter. Some of the major elements of computational complexity, uncertainty management, verification and validation, multi-echelon and multi-channel supply chain design, low carbon, pull economy, etc. are discussed in this chapter. Also, the objective in this chapter is to introduce and review the foundations based on which the surrogate model selection criteria is developed. Besides the underlying surrogate model approaches, methods and tools reviewed are classified in terms of concept, application to surrogate model process and value in design.

In Chapter 3, the theoretical foundations for designing simulation-based design processes are discussed. These foundations include existing design constructs using surrogate models, different surrogate models and different evaluation measures. Also explored in this chapter is regarding design process modelling. Foundational constructs are reviewed from this area. Relevant literature for each of these areas is referenced, discussed, and critically evaluated to show the appropriateness of use of these constructs for the design architecture developed in the dissertation. The literature review in Chapter 3 is used to identify availability, strengths, and limitations of these constructs in the context of managing complexity and uncertainty using approximation through surrogate modelling and becomes an essential component of theoretical structural validation. The relationship of these research efforts reviewed in this chapter with the constructs of the systematic approach developed in this dissertation is highlighted in Figure 3.1.

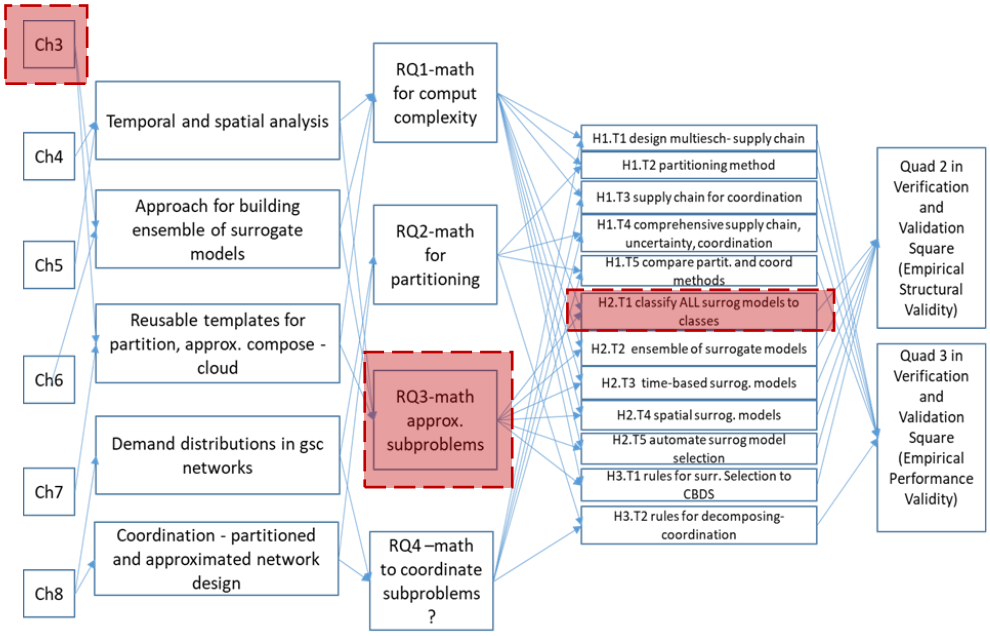


Figure 3.1. Relationship of research efforts with the constructs of the systems-based design architecture and connection between chapters of the dissertation.

In Section 3.1, a critical evaluation of the literature on surrogate modeling is provided. While in Section 3.2, design of experiment (DOE) methods are shown and different DOEs are compared, model choices of surrogate models, model fitting methods and their comparison are explained in Sections 3.4, 3.5 and 3.6 respectively. Results of critical evaluation of literature on surrogate modeling and the verification and validation of critical evaluation of literature on surrogate modeling are illustrated in Sections 3.7 and 3.8 respectively.

ABBREVIATIONS USED IN THIS CHAPTER

ANN	Artificial Neural Network
CCD	Central Composite Design
CFD	Computational Fluid Dynamics
CPU	Central Processing Unit
DOE	Design of Experiments

DST	Dempster–Shafer Theory
EA	Evolutionary Algorithm
FD	Factorial Design
FEA	Finite Element Analysis
FFD	Fractional Factorial Design
EMO	Evolutionary Multiple-Objective
DOE	Design of Experiments
GSME	Generalized Mean Square Error
KRG	Kriging
LS	Least Squares
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MAXE-CV	Maximum Absolute Cross-Validation Error
MSEGO	Multiple Surrogate Efficient Global Optimization
MEMO	Multimodal-based Evolutionary Multiple-Objective
NSGA-II	Non-dominated Sorting Genetic Algorithm II
OA	Orthogonal Array
PCA	Principal Component Analysis
POF	Pareto Optimal Front
PRESS	Predicted Residual Error Sum of Squares
PRS	Polynomial Response Surface
PSO	Particle Swarm Optimization
RBDO	Reliability-Based Design Optimization
RBF	Radial Basis Function
RMSE	Root Mean Square Error
RSM	Response Surface Models

SE	Square Error
SVM	Support Vector Machines
WAS	Weighted Average Surrogate
WLSR	Weighted Least Square Regression

In simulation-based realization of complex systems, we are forced to address the issue of computational complexity. One critical issue that must be addressed is the approximation of reality using surrogate models to replace expensive simulation models of engineering problems. In this section, we critically review over two hundred papers. We find that a framework for selecting appropriate surrogate modeling methods for a given function with specific requirements is lacking. To address this gap, we hypothesize that a trade-off among three main drivers, namely, size (how much information is necessary to compute the surrogate model), accuracy (how accurate the surrogate model must be) and computational time (how much time is required for surrogate modeling process) is needed. In the context of the hypotheses, we review the state-of-the-art surrogate modeling literature to answer the following three questions:

1. What are the main classes of the design of experiment (DOE) methods, surrogate modeling methods and model-fitting methods based on the requirements of size, computational time, and accuracy?
2. Which surrogate modeling method is suitable based on the critical characteristics of the requirements of size, computational time, and accuracy?
3. Which DOE is suitable based on the critical characteristics of the requirements of size, computational time, and accuracy?

Based on these three characteristics, six different categories for the surrogate models are framed through a critical evaluation of the literature. These categories provide a framework for

selecting an efficient surrogate modeling process to assist those who wish to select more appropriate surrogate modeling techniques for a given function. It is also summarized in Tables 3.4, 3.5, and Figure 3.5). Artificial neural networks, response surface models, and Kriging are more appropriate for large problems, less computation time and high accuracy, respectively. Also, Latin Hypercube**,** fractional factorial designs and D-Optimal designs are appropriate experimental designs.

3.1 Critical Evaluation of The Review Papers Written on Surrogate Modeling

Over the last few decades, computer simulation models, which are often used to represent physical problems via mathematical models and computer code, have begun to play a crucial role in engineering problems. Many simulation models are applied in various types of engineering problems, including optimization design, uncertainty design, reliability analysis, reliability-based design optimization (RBDO) and robust design. However, these computer simulations have a strong tendency to be computationally expensive due to their excessively detailed representation of real-world systems. Furthermore, many model-based engineering problems require simulation models to be run thousands of times to develop an appropriate solution, which also demands a high computational budget.

Surrogate models (sometimes called metamodels) are commonly used to replace expensive simulation models of engineering problems to mitigate the large computational budget in engineering design problems. One or more variables are altered, and then, the impact of this change is analyzed. Thus, researchers have developed experimental design methods as alternatives to physical experiments. These methods have been implemented as simulation-based designs to enhance the efficacy of these studies.

Various applications that use experimental design techniques in simulation-based design are statistically arguable. Because computer code is deterministic, the error of approximation is not caused by random effects. However, in physical experiments, random errors, such as measurement errors, can occur. Many researchers in computer-based design have eliminated this uncertainty and consider the error of approximation. Therefore, due to this difference between physical experiments and computer experiments, appropriately designed experiments considering both measurement and approximation errors are necessary for the efficient use of computer simulation and analysis to manage uncertainty. Hence, in this chapter, we review the literature regarding surrogate modeling and shed light on the state-of-the-art methods in this area of research. The intellectual questions that are posed and investigated in this chapter are summarized in Table 3.1.

Table 3.1. Questions investigated in this chapter.

Research Questions	Relevant Sections
1. What are the main classes of the design of experiment (DOE) methods, surrogate modeling methods and model-fitting methods based on the requirements of size, computational time, and accuracy?	3.2.2, 3.2.3, 3.5.1, 3.5.2
2. Which surrogate modeling method is suitable based on the critical characteristics of the requirements of size, computational time and accuracy?	3.3, 3.4, 3.5.2
3. Which DOE is suitable based on the critical characteristics of the requirements of size, computational time and accuracy?	3.5.1, 3.5.2

The abovementioned questions help us to identify more appropriate surrogate modeling process, including the selection of a DOE to produce data and subsequent use of a model to represent the generated data. Then, we need to create the best fit between the model and the data. Thus, section two is divided into three subsections based on this classification. In the first subsection (3.2), DOE methods are introduced and described. Then, before describing the different DOE techniques, we explain the problem of size and how to address this issue with the most well-known dimensionality reduction methods, such as principal component analysis, variable screening, partitioning, and Bayesian updating.

A broad range of DOE techniques are then reviewed. In the second subsection, an overview of different methods that can be used to represent the data generated by DOEs is provided. In this subsection, in addition to methods that have been used by researchers for a long time, such as response surface models, Kriging and neural networks, more recently developed methods, such as reference-point-based Nondominated Sorting Genetic Algorithm II (R-NSGA-II), are reviewed. In the third subsection, methods used for fitting the selected models to represent the data are briefly discussed. Finally, an approach for comparing different surrogate modeling methods is introduced based on these fitting indicators.

Although computers become faster and more powerful every day, the increasing computational complexity demand is still much higher than the increasing power and speed of computers. Additionally, serious difficulties in using computer simulations can occur due to their high fidelity regardless of how fast and powerful the computers used are. Even methods, such as parallel computing, which is a computation in which many calculations or executions of processes are carried out simultaneously, are not very helpful (Grama, 2013).

The most important disadvantage of parallel computing is that designing and producing shared-memory machines becomes increasingly difficult and expensive as the number of processors increases. A second disadvantage is the lack of scalability between memory and CPUs because adding more Central Processing Units (CPUs) can geometrically increase traffic on the shared-memory-CPU path and, in coherent cache systems, geometrically increase traffic associated with cache/memory management (Geist, 2000). Therefore, instead of using the original costly computer analysis, statistical approximation methods are used to build surrogates for expensive, complicated computer-based code to efficiently manage computational complexity. We use these surrogate models in the initial phases of the design to explore the solution space and develop a set of solutions.

There are many types of surrogate modeling techniques for approximating reality. Response surface methodology (RSM) (Box et al., 1978b) and artificial neural networks (ANN) (Cheng and Titterington, 1994) are methods used for creating fast estimations of intricate mathematical models. Thus, many people have used Kriging instead of complicated mathematical models, such as (Sacks et al., 1989). Recently, other statistical methods, such as inductive learning (Simpson et al., 2001), adaptive learning (Picheny et al., 2010), and different machine learning methods, including genetic algorithms (Deb et al., 2017; Li et al., 2017) and support vector machines (SVM) (Brereton and Lloyd, 2010; Gunn, 1998; Viana et al., 2013), have attracted attention.

The principal challenge faced by all designers who use metamodels is as follows: what is **more** appropriate approach for the surrogate modeling process? What are the circumstances in which one approach can be **more** appropriate? In several papers, the authors use one surrogate modeling method for a task. Simpson et al. (Simpson et al., 1998) compared Kriging and polynomial regression to address an optimization design problem with three design variables. In (Giunta et al., 1994), another comparison between Kriging and regression functions was performed. Additionally,

(Varadarajan et al., 2000) compared polynomial regression to the ANN technique to address an engine design problem including nonlinear thermodynamic performance. In (Yang et al., 2000), four surrogate modeling techniques, including ANN and stepwise regression, were compared in a vehicle accident investigation. Although problem-specific factors must be considered in the selection of a surrogate modeling approach, we believe that a comprehensive study comparing and classifying the different techniques and a well-an organized process for examining the relative advantages of different methods remain lacking.

Multiple factors in a particular problem, including linearity/nonlinearity, the size of the problem, the desired level of accuracy, the speed of the process, the required amount of information, the size of the sample, and the availability of convenient computer code, impact the appropriateness of a surrogate modeling technique. We believe that using one measure of merit, such as accuracy, for comparing surrogate models in surrogate modeling is not efficient, and multiple measures, such as a combination of accuracy, convenience, robustness, efficiency, simplicity, and model transparency, must be considered as mentioned in (Jin et al., 2000). Therefore, we contend that to select one metamodeling technique for a task, knowledge regarding the characteristics of various surrogate modeling techniques using various modeling measures is needed.

In this chapter, a comparative study investigating the characteristics of several surrogate modeling techniques is conducted with a focus on a critical review of the surrogate modeling literature using different modeling criteria to provide guidance to engineering designers. The tools, methods, and techniques used in the surrogate modeling process are discussed in detail here. The surrogate modeling process has three main steps, including (a) selecting a DOE to generate data, (b) selecting a model to describe the data, and (c) fitting the model to the observed data. In section

there are three main subsections based on the three principal steps of the surrogate modeling process, including a review of the methods used for the design of experiments (DOEs) and their evaluation, analysis of model selection and assessment, and model fitting and evaluation.

In this section, an overview of the reviews of the surrogate modeling literature is presented. A summary of previous review papers is provided in Table 3.2. In (Simpson et al., 2001), the authors perform a broad survey of metamodeling uses in aerospace and mechanical systems; in (Barthelemy and Haftka, 1993), a review of metamodeling applications in structural optimization is carried out. The use of surrogate modeling in multidisciplinary design optimization is discussed in (Sobieszczanski-Sobieski and Haftka, 1997). A review by (Viana, 2008) advises against employing a single surrogate model and alternatively proposes that multiple surrogate models are appropriate and even can be used in association with one another. The multiple surrogate model methods are useful in water resource applications (Chen et al., 2006).

Similarly, to radial basis function (RBF) models, the minimum number of design points needed to fit a Kriging model is the number of coefficients in the polynomial augmenting the approximation. Kriging users only need to specify the lower and upper bounds of the correlation parameters, although the appropriate bounds can be difficult to specify. Kriging correlation parameters can be interpreted to some extent in that large values of a dimension indicate a highly nonlinear function in that dimension, while small values indicate a smooth function with limited variation (Maier and Dandy, 2000).

(Simpson et al., 2008) report that the prevalent theme in six highly cited metamodeling (or design and analysis of computer experiments) review papers is indeed the high cost of computer simulations. Global optimization algorithms based on response surface models, such as EGO

(Jones et al., 1998), GMSRBF, MLMSRBF (Regis and Shoemaker, 2007), and Gutmann's method (Gutmann, 2001), and uncertainty analysis algorithms, such as ACUARS (Mugunthan and Shoemaker, 2006) and RBF-enabled MCMC (Bliznyuk et al., 2008), can help circumvent the computational budget limitations associated with computationally intensive simulation models. Therefore, surrogate modeling may only be beneficial when the simulation model is computationally intensive, justifying the expense of performing a second level of abstraction (reducing model fidelity), which typically leads to analyses with reduced accuracy. However, even though (Jones, 2001) and (Simpson et al., 2008) both note that surrogate modeling has more advantages than simply reducing computational time, reviewing other possible motivations for surrogate modeling is beyond the scope of this chapter.

The literature regarding response surface surrogates for engineering design optimization problems are also reviewed in (Simpson et al., 2001) and (Wang et al., 2014). The literature regarding response surface surrogate modelling is summarized in a discussion panel held at the 9th AIAA/ ISSMO Symposium on Multidisciplinary Analysis and Optimization (Simpson et al., 2004). The literature regarding response surface modelling and motivations from a historical perspective is evaluated in (Simpson et al., 2008) with a focus on the appeal of lower-fidelity physically based surrogate modelling.

Recent advances in surrogate modelling, including advances in lower-fidelity physically based surrogates in the field of optimization is reviewed in (Forrester and Keane, 2009). Special journal issues focusing on surrogate modelling summarize the first and second International Workshops on Surrogate Modelling and Space Mapping for Engineering Optimization (Bandler and Madsen, 2001; W. Bandler et al., 2008)). Another special issue publication regarding surrogate modelling

is a thematic journal issue focusing on surrogate modelling for sensitivity analyses and the reduction of complex environmental models (Ratto et al., 2012).

Additionally, there are more specific review papers focusing on specific tools/strategies involved in surrogate modelling. Kleijnen [2009] reviews Kriging and its applications in response surface surrogate modelling. Multiple function approximation models acting as response surface surrogates are reviewed and compared in (Jin et al., 2001) and (Chen et al., 2006). In another study, response surface surrogate modelling used with evolutionary optimization algorithms (Jin, 2005). Gradient-enhanced metamodels, which involve function gradients that use common auxiliary information and are useful for predicting functions based on locally changing behaviors is reviewed in (Laurent et al., 2017).

These authors review the main metamodels using function gradients in addition to function values. Additionally, structural crashworthiness in design optimization is reviewed in (Fang et al., 2017). Surrogate modelling has become increasingly popular over the last decade within the water resources community, which is consistent with the increasing utilization of metamodels in the scientific literature since 1990 as documented by Viana and Haftka (Viana, 2008).

As shown in Table 3.2, no survey analyzes the characteristics of more appropriate surrogate model for a design problem. Additionally, a comprehensive set of criteria for identifying appropriate surrogate models is lacking. Furthermore, no authors have discussed the automation of the surrogate modeling process. We aim to address these gaps in this chapter.

Table 3.2. Summary of previous review papers

Paper	Focus	Type of Problem	Types of Metamodels
(Fang et al., 2017)	Design optimization for structural crashworthiness	Engineering design	Polynomial response surface (PRS) model, radial basis function (RBF) model, Kriging (KRG) model, artificial neural network (ANN) model, comparison of different surrogate models, ensemble of surrogates
(Laurent et al., 2017)	Gradient-enhanced surrogate models	Numerical comparisons of metamodels carried out for approximating analytical test functions	Classical, weighted and moving least squares, Shepard weighting functions, and kernel-based methods that are radial basis functions, Kriging and support vector machines
(F. A. C. Viana, and R. T. Haftka, 2008)	Multiple surrogate models	Engineering design	Kriging, radial basis neural networks, linear Shepard and support vector regression
(Chen, Tsui, Barton, & Meckesheimer, 2006)	Wastewater treatment stochastic dynamic programming (SDP)	Aerospace engineering: engineering design and optimization; electrical engineering: CAD/CAM modeling and optimization; chemical engineering: optimization of a continuous-stirred tank reactor; continuous-state stochastic dynamic programming	Two-dimensional response, surfaces and one ten-dimensional surface, polynomial response surface models, MARS, spatial correlation models, OA designs, regression trees and related methods, least interpolating polynomials, ANNs, RBFs

Paper	Focus	Type of Problem	Types of Metamodels
(Kleijnen, 2009)	Extension of Kriging to random simulation and discussion of bootstrapping to estimate the variance of the Kriging predictor	In addition to classic one-shot statistical designs, such as Latin hypercube sampling, sequentialized and customized designs for sensitivity analyses and optimization are reviewed	Kriging and its applications in response surface surrogate modeling, such as RBF models
(Maier & Dandy, 2000)	Review of modeling issues and applications	Neural networks for forecasting water resource parameters	ANN
(Jones, Schonlau, & Welch, 1998)	Efficient global optimization	Balancing the requirement to use the approximating surface (using sampling when it is reduced) with the requirement to enhance the estimation (by sampling when forecast error can be significant)	Response surface methodology
(Sacks et al, 1989)	Many scientific phenomena are currently investigated by sophisticated computer models or code. Usually, the computer models are computationally expensive	Design and analysis of computer experiments	RSM, neural networks, function bounds

Paper	Focus	Type of Problem	Types of Metamodels
	to run, and the mutual objective of the experiment is to fit a cheaper predictor of the output to the data		
(Sobieszczanski-Sobieski and Haftka 1989)	Primary challenges in MDO include computational expense and organizational complexity	Engineering design	Kriging, radial basis neural networks, linear Shepard
(Barthelemy and Haftka, 1993)	Applications of nonlinear programming methods to large structural design problems could be cost-effective if suitable approximation concepts are introduced	Design and analysis of computer experiments	RSM, neural networks, function bounds
(Regis & Shoemaker, 2007)	Efficient global optimization	Groundwater bioremediation models	Response surface surrogates GMSRBF and MLMSRBF

Paper	Focus	Type of Problem	Types of Metamodels
(Mugunthan & Shoemaker, 2006)	Uncertainty analysis algorithms	Assessing the impacts of parameter uncertainty in computationally expensive groundwater models	ACUARS and RBF-enabled MCMC
(Bliznyuk et al., 2008)	Basic sequential framework	Automatic calibration and Bayesian uncertainty analysis of an environmental model	Radial basis functions
(Jones, 2001)	Adaptive-recursive framework	Local optimization	Response surface surrogate modeling, Kriging
(T. Simpson et al., 2008)	Multi-level/multi-fidelity approximations and ensembles of metamodels	Availability of metamodels in commercial software for design space exploration and visualization	Two-dimensional response, surfaces and one ten-dimensional surface, MARS, spatial correlation models, OA designs
(T. W. Simpson et al., 2001)	Computer-based engineering design	Approximation of deterministic computer analysis codes	Two-dimensional response, surfaces and one ten-dimensional surface, polynomial response surface models, MARS, spatial correlation models, OA designs, regression trees and related methods, least interpolating polynomials, ANNs, RBFs
(Wang et al., 2007)	Model approximation, design space exploration,	Finite element	Polynomial (linear, quadratic, or higher), splines (linear, cubic, and NURBS), multivariate adaptive regression splines (MARS), Gaussian

Paper	Focus	Type of Problem	Types of Metamodels
	problem formulation, optimization support	analysis (FEA) and computational fluid dynamics (CFD)	process, Kriging, radial basis functions (RBF), least interpolating polynomials, artificial neural network (ANN), knowledge base or decision tree, support vector machine (SVM), hybrid models
(Simpson et al., 2004)	Approximation methods in multidisciplinary analyses and optimization	Design of experiments versus design and analysis of simulations or computer experiments, reflecting experimental results, and data from approximation models, capturing uncertainty with approximation methods, addressing problems with numerous variables	Surfaces and one ten-dimensional surface, MARS, spatial correlation models, RBFs
(Forrester & Keane, 2009)	Efficient global optimization	Advances in lower-fidelity physically based surrogates in the field of optimization	Polynomials, moving least squares, radial basis functions, radial basis function models of noisy data, Kriging, universal Kriging, blind Kriging with noisy data, support vector, support vector predictor
(Bandler & Madsen, 2001; W. Bandler, Koziel, & Madsen, 2008)	Editorial-surrogate modeling and space mapping	Engineering optimization	Artificial neural network approaches, Kriging, quadratic response surfaces, and approaches based on splines

Paper	Focus	Type of Problem	Types of Metamodels
(Ratto, Castelletti, & Pagano, 2012)	Tools involved in surrogate modeling	Reduction and sensitivity analyses of complex environmental models	RSM, Kriging, RBF, ANN
(Y. Jin, 2005)	Comprehensive survey of fitness approximation in evolutionary computation	Response surface surrogate modeling if used with evolutionary optimization algorithms	Response surface surrogate, evolutionary algorithms
(Chen et al., 2003)	Primarily on the task of metamodeling, which is driven by the goal of optimizing a complex system via a deterministic simulation model. Also, the case of a stochastic simulation, and examples of both cases	Applications in electrical engineering, chemical engineering, mechanical engineering, and dynamic programming	Response surfaces, Kriging, regression splines, regression trees, neural networks, orthogonal arrays, Latin hypercubes, number-theoretic methods

3.2 Design of Experiments (DOEs)

In this section, we explain the DOEs, discuss the different types of DOEs and perform a comparison of the types; for any comparison, some indices or metrics are needed; thus, we provide an overview of existing and most frequently used evaluation methods used as metrics for assessing DOEs. To efficiently use computer-based code, constructing experimental design methods appropriately is essential. Thus, engineers have attempted to observe the effects of changes in one parameter on other parameter(s) for a long time. Additionally, in another method, engineers have examined large factor sets through a systematic or random process to compare the factor sets simultaneously. A plethora of different approaches can be used to design and apply such experiments to the creation of surrogate models. In this section, we review the various types of DOEs and measures of merit used to evaluate the DOEs.

3.2.1. The problem of size in surrogate modeling

In the surrogate modeling process, addressing the problem of size, which is related to the number of variables, is among the greatest challenges. Hence, addressing this issue first is essential as it directly influences the entire process of surrogate modeling and particularly the DOE. Because the computational time is directly related to the number of factors, an effective strategy for managing the computational time is to reduce the problem dimension. Thus, before introducing different DOEs for the generation of design points, which are a combination of design variables (factors), reviewing some of the most frequently used dimensionality reduction tools is helpful.

3.2.2. Principal component analysis as a dimensionality reduction method

In principal component analysis (PCA) technique, we use an orthogonal conversion through to transform an observed variable set which is correlated with an independent linear set of variables called principal components (PCs) (Jolliffe, 2005). The number of PCs is always smaller than the number of initially observed design variables. In this conversion, the highest variance is assigned to the first PC, and the subsequent variances are second highest, third highest, etc. on an orthogonal basis (Jolliffe., 2002). PCA eigenvalues to determine lower-dimensional sets of variables which retain the maximum possible information (Gao et al., 2018). PCA is appropriate for reducing very high dimensionality in the solution space. For example, thousands of variables could be reduced to less than a dozen.

3.2.1. Variable screening as a dimensionality reduction method

Screening the variables refers to discovering significant variables in a set of random variables. In this definition, the word "significant" has various implications based on the problem investigated (He et al., 2013). These types of experiments are used to identify the most notable factors and then reduce the number of variables to construct higher-order models. There are several types of variable screening methods; for example, (Koch et al., 1999) use a Pareto method to examine the outcomes of an experimental design in grading the significance of the factors in each response. As a DOE method, the 2-level fractional factorial design has been applied to screen 26 control and noise factors including 64 designs and one center point. At the end of the process, 18 factors remain; 14 factors are control factors, and four factors are noise factors. Additionally, in (Cho et al., 2014), the authors introduce a useful approach for variable screening in surrogate modeling with fractional output variance based on univariate dimension

reduction. In this variable selection process, the authors chose variables that yield higher output variance as vital factors. In this industrial study, the authors select 14 factors from a set of 44 design variables and consider 11 constraints for the output variance.

3.2.2. Partitioning as a dimensionality reduction method

Partitioning the problem and multiple-dimensional or multiple-level optimization represent other approaches for addressing dimensionality. A comprehensive survey of methods used to decompose a problem into more manageable partitions is available in (Lewis and Mistree, 1998). In addition, in (Sobieszczanski-Sobieski and Haftka, 1997), several multidisciplinary design optimization techniques for framing and solving partitioned problems concurrently are introduced. Moreover, Balling and Wilkinson in (Balling and Wilkinson, 1997) classify hierarchical partitioning methods into the following three classes: i) single-stage problems, ii) concurrent subproblems, and iii) collaborative techniques. The most distinctive feature of the single-stage methods is the single discipline feasible exploration and design. In the 2nd type of problem partitioning, we address partitions independently and concurrently during the solution process. Then, the compatibility of the solutions for each subproblem is assessed (Koch et al., 1999).

3.2.3. Bayesian updating as a dimensionality reduction method

If several simulations can be performed and many design variables must be included in the surrogate model, Bayesian updating is useful and can overcome the difficulties causing classical stochastic optimization to fail (Cheng and Currie, 2004). Thus, in the case of numerous design variables and a complex simulation model, only limited runs can be

implemented, and approximating all coefficients of the surrogate model using classical tools, such as maximum likelihood, may be impossible (Liang and Mahadevan, 2016). In such cases, Bayesian updating has an enormous benefit.

In Bayesian updating, the initial probability distribution $f(x)$ is the first estimate of the probable value of the parameters that can achieve a feasible solution for starting the simulation runs. Then, the start point value is gradually improved, and estimates of the distribution of x become more accurate as the simulation runs proceed (Beck and Au, 2002). Therefore, the prior lack of information regarding the parameters is addressed by allocating initial distributions to the parameters and using experimental data. We revise this lack of information and update their distributions (from initial to subsequent) according to the data observed using Bayesian updating analysis (Ghanem and Doostan, 2006; Kennedy and O'Hagan, 2001).

There are two types of model updating strategies, namely, bias-correction and calibration, which differ in formulation and the technique used to obtain a solution (Xiong et al., 2009). We can use bias-correction if the calibrating parameters of the model cannot enhance accuracy (Easterling and Berger, 2002; Hasselman et al., 2005). We can also use bias-correction to capture the probable model error caused by inaccurate modeling; for instance, a linear method can be used to model nonlinear behavior that usually cannot be captured by other methods (Kennedy and O'Hagan, 2001). Additionally, using the calibration approach, typically, two types of inputs serve as controllable variables and uncontrollable variables that are unchanged during the simulations. However, the uncontrollable variables should be calibrated (Trucano et al., 2006).

3.2.4. The main types of experimental designs

Numerous surrogate modeling methods are available, including factorial designs, fractional factorial designs (Gunst and Mason, 2009; Montgomery, 2017), central composite designs (Montgomery, 2017), Box-Behnken (Box and Behnken, 1960), orthogonal arrays (Hedayat et al., 2012), (Dey, 1985; Montgomery, 2017), D-optimal (Myers et al., 2016), Placket-Burman (Gustafsson et al., 2013), hexagon (Montgomery, 2017), hybrid (Myers et al., 2016), Latin Hypercube (McKay et al., 1979; Montgomery, 2017), random selection, grid search (Barthelemy and Haftka, 1993; Corchado et al., 2007; Güntert et al., 1998; Hsu et al., 2003), etc. We review some of these methods.

3.2.5. Factorial designs

Factorial designs are the most fundamental experimental designs. In this experimental design method, the number of factor levels identifies the number of design points. Two main factorial designs (FD) are 2^k (for two levels and k factors) (Box and Hunter, 1961) and 3^k. 2^k designs are used to evaluate main effects and interactions and 3^k design are used to assess the quadratic core interactions and impacts. In FD, factor refers to a design variable that may affect the other design variable, that is, the response variable. Level describes the discrete possible values that each factor can take; in DOEs, to ensure that the process is reasonable, we merely consider limited possible values for factors (Box et al., 1978a). For instance, in a two-factorial design, there are three (low-medium-high) levels, leading to 3^2=9 different design points.

3.2.6. Fractional factorial designs (FFDs)

The number of runs needed for a full factorial design increases exponentially and rapidly exceeds the available resources of most experimenters by increasing the factors through a 2^k factorial design, e.g., a full iteration of the seven factors requires 128 runs. Under such circumstances, the engineering designer can obtain knowledge regarding the critical impacts using only a small portion of the full design (Montgomery, 2017). Fractional factorial designs (FFD) utilize the particular specifications of the design to decrease the size of an experiment and limit the loss of critical knowledge that might be ignored by not performing a wide-range study simultaneously involving probable groups of the levels. Screening is among the principal uses of this type of DOE (Gunst and Mason, 2009).

In experiments commonly using this type of DOE, the number of required designs is 2^{k-p}, such that $\frac{1}{2^p}$ is the fraction. Typically, we use a $2k$ design to recognize and screen the critical design variables. In addition, we can use $2k$-p for this purpose. The rare effect principle can occur if numerous factors lead a system predominated by significant effects and interactions with low order. Therefore, to screen the factors, 2-level designs can be used (Montgomery, 2017).

3.2.7. Central composite designs

Central composite designs (CCDs) are typical second-order designs used to explore higher-order impacts, mainly quadratic effects, that require many design points. CCDs are 2-level FDs with 2^k star points for a design with k factors. Each star point is placed at a position with a $\pm\beta$ distance from each factor. The distance between the center and each star in the design solution

space is $|\beta|>1$ if the distance between the center and a factorial point is ±1 unit for each factor. For example, if there are three factors, by using $\beta=1$, the star points are placed in the middle of the faces of a cube (Montgomery, 2017).

3.2.8. Orthogonal arrays

Orthogonal arrays (OAs) are the experimental designs recommended by Taguchi. The designs represent a type of FFD that is used primarily in 2 or 3 levels (Owen, 1992). These designs are built to decrease the number of essential design points; for instance, with 4, 12 and 16 design points, it is possible to assess 3, 11 and 15 variable/impacts by second level L4, L12, and L16 arrays. This type of design is also known as a Plackett-Burman design (Hedayat et al., 2012).

3.2.9. Space-filling designs

In the early stages of design very little is known about the mathematical models of the design and exploring a design space identifying the appropriate surrogate model can be impossible. Space filling design allow users to explore all regions of the design space equally. Space-filling designs are used to assist designers in very early stages of design (Montgomery, 2017).

3.3 Comparison of Different DOEs: Evaluation Metrics

Selecting the appropriate design is critical for efficient experimentation, that is, balancing between obtaining knowledge about the relationships among factors, the response variables and computational time. Thus, after providing an overview of the available and useful DOEs, in this subsection, we review the criteria used to evaluate and compare DOEs. Then, we

critically discuss the utility and usefulness of each DOE that is introduced in Section 3.2.2 based on knowledge regarding the DOE and its cost and computational time. In this subsection, we review some measures of merit used to evaluate the appropriateness of a chosen DOE for generating design points.

3.3.1. Unsaturated, Saturated, and Supersaturated Designs

The most significant challenge in DOEs is the size of the problem. Unsaturated designs are the most common types of designs in which the number of design points is twice the number of factors. In saturated DOEs, the number of design points and number of factors are equal (Chen and Lin, 1998). However, the design points and factors have different characteristics; for example, a fractional factorial design that is saturated enables us to predict significant effects in an unbiased way with small size and variance (Montgomery, 2017). There are many examples of saturated designs, and OAs, which were proposed by Taguchi, and two-level Placket-Burman designs are the most frequently used designs (Simpson et al., 2001).

Supersaturated designs represent a particular type of FD and are typically used if the factors outnumber the observations (Nguyen, 1996). Building this sort of design by a random process is recommended (Kathleen and Cox, 1962; Lin, 1993). In this type of design, we do not have an adequate number of runs to approximate significant effects. We use this type of design in the screening process to identify principal and sparse factors through a process which requires limited computation(Phoa et al., 2009). Frederick and co-authors in (Phoa et al., 2009) studied supersaturated designs using the Dantzig selector and found that this variable selection method is highly effective in predicting the size of the model. If the essential runs of the simulations

are not extremely expensive, using unsaturated designs for predictive functions is recommended (Simpson et al., 2001).

3.3.2. Minimum bias

All approximators must be unbiased. Thus, the long-term expected value or average should be the same as the estimated parameter (Karson et al., 1969). Unbiasedness is necessary; however, this feature alone is not always sufficient to ensure that an approximator is a good approximator (Montgomery, 2017).

3.3.3. Minimum variance

The variance of the coefficient predictions in a first-order model is minimal if we build it as follows in Equation 3.1:

$$\sum_{u=1}^{N} x_{iu}{}^{2} = N \qquad \text{Equation 3.1}$$

where the approximation variance ($\hat{y}$) is an unmoving variance, and the distance from the design center is constant (Simpson et al., 2001).

3.3.4. Rotatability

Rotatability is an evaluation metric used for second-order models. For a 2nd-order model, we should create appropriate approximations for the entire solution space. One of the characteristics that shows that a model is "appropriate" is a rationally constant variance at the approximated response (Khuri, 1988). Box and Hunter recommended that a 2nd-order RSM must have rotatability. Thus, the variance of the approximator is the same at all points *x* that

are equidistant from the center of the design. Thus, the deviation of the approximated response is consistent on spheres (Box and Hunter, 1957).

This feature in a design problem means that the variance remains constant as the design is rotated around the center (0, 0, . . . , 0), and thus, this design is called a rotatable design (Draper and Guttman, 1988). Rotatability is a rational foundation for selecting a response surface design. RSM is used to create an approximation and the place of approximation is unknown before implementing the experiment, using a design that creates equally accurate approximations in the estimation in all directions is reasonable (Khuri, 1988).

3.3.5. Orthogonality

If the summation of the products of the N design points for each x_i and x_j is zero (as shown below), the design is orthogonal.

$$\sum_{u=1}^{N} x_{iu} x_{ij} = 0 \qquad \text{Equation 3.2}$$

Thus, if the aggregation of the products of the corresponding components of two vectors of the same length is zero, the design is orthogonal. Additionally, if the impacts of any factor balance out the impacts of other factors (sum to zero), the experimental design is orthogonal (Gunst, 1996). By performing a comprehensive review of designs of experiments and a comparison of different DOEs and evaluation metrics of DOEs, we determined that the critical characteristics that should be considered by a designer/decision-maker in identifying the most appropriate surrogate modeling method are the problem of size, the accuracy level needed for the output of the surrogate model and the computational time (Xue et al., 2013). In this section,

we compare the utility and appropriateness of the previously introduced DOEs based on the required numbers of factors and design points, the computational time and the expected accuracy level.

3.4 Model Choice in Surrogate Modeling

In this subsection, we summarize some of the models that can be used to represent the data generated in the previous step with DOEs.

3.4.1. Response surfaces

If we assume that *x* is an independent vector of factors and *y* is the vector of response, then the impact of *x* on *y* and their relationship is:

$$y = f(x) + \varepsilon \qquad \text{Equation 3.3}$$

where ε denotes the normally distributed random error with a mean of zero and a standard deviation of 0 (Box and Draper, 1987). Because the actual response surface function is indefinite, instead of *f(x),* a new response surface *g(x)* is built and used as a surrogate for *f(x).* Thus, $\hat{y} = g(x)$ yields the estimated values of *f(x)* (Khuri and Mukhopadhyay, 2010). Because knowledge regarding the relationship between the response variables and factors in most RSM problems is limited, we should first perform an appropriate estimation of the correct function of the response and factors (Deaton and Grandhi, 2014). The approximating function is the 1st-order model if we can appropriately demonstrate the actual model by a linear function of the factors.

$$y = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_i x_i + \epsilon$$ Equation 3.4

However, sometimes, we need to use a higher-order polynomial, such as a 2nd-order model (see Equation 3.5), if there is curvature in the problem.

$$y = \alpha_0 + \sum_{i=1}^{k} \alpha_i x_i + \sum_{i=1}^{k} \alpha_{ii} x_i^2 + \sum\sum_{i<j} \alpha_{ii} x_i x_j + \epsilon$$ Equation 3.5

Regardless of whether RSM uses either a linear or a higher order function, a polynomial model is not always a rational estimation of the actual functional relationship throughout the region of the factors. These models typically perform well in a relatively small space (Montgomery, 2017). RSM is very easy to use and very convenient.

3.4.2. Kriging

Kriging is a surrogate modeling technique introduced as a mixture of a polynomial model of *x*, that is, *f(x)*, and localized deviations of *x*, that is, *Z(x)*, as shown in Equation 3.6.

$$y(\boldsymbol{x}) = f(\boldsymbol{x}) + Z(\boldsymbol{x})$$ Equation 3.6

where *Z(x)* refers to the concept of a normally distributed Gaussian random process with a mean of zero, variance ^{2}, and non-zero covariance. In Equation 3.6, *f(x)* is a polynomial function of an RSM that delivers a 'global' model and is often a fixed function (Stein, 2012). Kriging is a Gaussian process regression and is a technique of interpolation. The interpolated values are modeled by a Gaussian process using information about prior covariances (Gano et al., 2006).

3.4.3. Radial basis functions (RBF)

This technique is a mathematical function that accepts real values, and its value is calculated based on the distance between the origin and each point (Shan and Wang, 2010). Alternatively, the distance between the center point and each point can be used, as shown in Equation 3.7.

Q(x) = *Q* ($\|x\|$) (distance from the origin point); *Q (x, c)* = *Q* ($\|x - c\|$) (distance from the center point) Equation 3.7

In surrogate modeling, the integration of these functions is used to estimate complicated mathematical functions. We can use these functions to construct surrogate models, as shown in Equation 3.8.

$$\hat{y}(x) = \sum_{i=1}^{M} r_i Q(\|x - x_i\|) \quad \text{Equation 3.8}$$

where the surrogate function $\hat{y}$ refers to the integration of *M* radial basis functions, and each function is linked to a distinct x_i and has a weight of w_i (Broomhead and Lowe, 1988).

3.4.4. Inductive learning

Machine learning is categorized in five significant paradigms: genetic algorithms, case-based learning, neural networks, analytic learning and inductive learning (Dumais et al., 1998). Inductive learning is a machine learning paradigm that is similar to metamodeling and regression. In this method, the necessary rules are derived from examples; these condition-action rules are used to divide the data into distinct classes and are the principal modeling

concepts and can simplify the interpretation by incorporating these into decision trees (Michalski, 1983).

We train the data in (x_i, y_i) pairs, where x_i is the feature vector values, and each y_i is the related result value. In the inductive learning technique, both the features and results can assume real values. However, the data can be fitted well if discrete values are used. Therefore, converting real values into discontinuous representations is a possibility. Then a decision tree can be created by training algorithms as soon as the data are gathered, which is possible after selecting appropriate features for separating and classifying the data. Choosing the features in a systematic manner that requires the least amount of information instead of chance is always recommended (Witten et al., 2016)**.**

The parameters represent undetermined components that must be learned from the data. In linear regressions, the coefficients θ are the parameters. To represent the parameters, we typically use θ (since there are numerous parameters in a model, this is a loose definition) (Madala and Ivakhnenko, 1994).

3.4.5. Boosted trees

Friedman introduced boosted trees or a gradient boosting machine as a method of supervised learning (Friedman, 2001). The boosted trees method is used in supervised learning problems in which multiple features x_i are available in a training dataset to estimate the response variable y_i. In supervised learning, the model typically denotes the mathematical construction of how to estimate y_i based on x_i. For instance, a linear model is very common, and the estimation is given by $\hat{y} = \sum_j \alpha_j x_{ij}$, which is a linear combination of weighted input features.

The approximation value can have different explanations based on the problem. For instance, to obtain the probability of a positive group in a logistic regression, logistic transformation can be used, and a ranking score can also be used to rank the output (Madala and Ivakhnenko, 1994). Boosted trees produce a prediction model in the form of an ensemble of weak prediction models that are typically decision trees. The boosted trees method is used to build models in a stage-wise fashion, which is similar to other boosting methods, and generalizes the models by allowing the optimization of an arbitrary differentiable loss function (De'Ath, 2007).

Boosted trees represent a tool for surrogate modeling and predictive data mining and have some specific properties. Boosted trees inherit the desirable features of trees while eliminating many undesirable features. The most desirable feature is robustness (La Fuente and Andres, 2016; MacCalman et al., 2016). All boosted trees are invariant under all (strictly) monotone transformations of individual input variables. For example, using x_j, $\log x_j$, e^{xj}, or x^{α}_j as the j^{th} input variable yields the same result. Thus, the need to consider input variable transformations is eliminated. Sensitivity to long-tailed distributions and outliers is also eliminated (Jagadeesh et al., 2016).

Boosted trees are entirely robust against outliers in the output variable *y*. Boosted trees also have a fair measure of robustness against output outliers (Lemercier et al., 2012). Internal feature selection is another benefit of decision tree conjecture. Trees are generally reasonably robust to the addition of input variables that are not relevant (Sim et al., 2018). Additionally, tree-based models address missing values in a unified and elegant manner (Song et al., 2018);

studying external imputation schemes is not necessary. Boosted trees also inherit these properties.

The principal disadvantage of single tree models is inaccuracy (Friedman, 2001). Inaccuracy is a result of the coarse characteristics of their piecewise continuous estimates, particularly for smaller trees. For larger trees, a major disadvantage is uncertainty, and the fact that they include predominately high-order interactions. These disadvantages are mitigated by boosting (Burnham and Anderson, 2003; Han et al., 2011; Vapnik, 2013; Witten et al., 2016). Boosted trees enhance stability by using small trees and averaging many of these trees. The interaction level of boosted tree approximations is adequately controlled by limiting the size of the individual constituent trees.

The most significant advantages of single tree models are interpretability, whereas boosted trees are thought to lack this feature (Duda et al., 2012). Small trees can be easily interpreted; however, due to instability, such interpretations should be treated with caution. The interpretability of more massive trees is questionable (Ripley, 2007). Another drawback of boosted trees is that they overfit their training sets and have a very high variance. Attempts have been made to use of random forests to overcome this drawback.

3.4.6. Random forests

Random decision forests or merely random forests are ensemble learning techniques for regression, classification, and other tasks that perform by building an aggregation of decision trees at the training stage and outputting the group that is the mode of the groups (classification)

or mean estimation (regression) of single trees (Ho, 1995, 1998). Random decision forests correct for decision trees' habit of overfitting to their training set (Friedman et al., 2001).

The first algorithm used for random decision forests was created by Tin Kam Ho (Ho, 1995) using the random subspace method (Ho, 1998), which, in Ho's formulation, was used to apply the "stochastic discrimination" approach to the classification proposed by Eugene Kleinberg (Kleinberg, 1990, 1996; Kleinberg, 2000). An expansion of this method was introduced by Leo Breiman (Breiman, 2001) and Adele Cutler (Cutler et al., 2012), and "Random Forests" is their trademark. The extension combines Breiman's "bagging" idea and the random selection of features first introduced by Ho (Ho, 1995) and subsequently developed independently by Amit and Geman (Amit and Geman, 1997) for building a controlled variance ensemble of decision trees.

Decision trees are a popular approach for many machine learning works. Tree learning is invariant under scaling, and various other conversions of feature values are robust to the insertion of irrelevant features and generate inspectable models. However, these models are rarely accurate (Friedman et al., 2001). Mainly, trees that are grown very deep can learn extremely abnormal patterns. Although these trees have considerable variance, they have a low bias because they overfit their training sets (Krauss et al., 2017). These trees can average multiple deep decision trees that have been trained on various parts of the same training set to decrease the variance (Friedman et al., 2001). However, while this approach is associated with a small increase in bias and some loss of interpretability, it generally dramatically boosts the performance of the final model.

3.4.7. Adaptive learning/active learning

Adaptive learning, sequential sample selection/design or active learning is a particular machine learning technique that is semi-supervised, and its algorithm can refer to the information source interactively to achieve the favored outcomes for a newly generated data set (R., 1970). This approach is also called an optimal design of experiments and is often used if many data points are unlabeled and/or cannot be labeled manually due to cost. In such cases, active learning is instrumental in collecting data; although the computation is very expensive, identifying the precise distribution of the data is nontrivial is in the initial stages of design, and accordingly, an iterative process is needed to select the data points (Gorissen et al., 2010). In active learning, there is a function for minimizing the number of samples chosen during the iterations and maximizing the knowledge attained by each step (Gorissen et al., 2010).

3.4.8. Hyperdimensional performance models

Hyperdimensional Performance Models are another type of surrogate models which are introduced by Turner in 2005. The distinct feature of Turner's proposed method is that surrogate models are constructed from a wide variety of mathematical basis functions but Hyperdimensional Performance Models (HyPerModels) are derived from Non-Uniform Rational B-splines (NURBs). Based on Turner's results, this can cause many unique advantages when compared to other surrogate modeling approaches. NURBs are defined by a set of control points, knot vectors and the NURBs orders, resulting in a highly robust and flexible curve definition that has become the de facto computer graphics standard. The defining components of a NURBs HyPerModel can be used to define adaptive sequential sampling algorithms that allow the designer to efficiently survey the design space for interesting regions.

The data collected from design space surveys can be represented with a HyPerModel by adapting NURBs fitting algorithms, originally developed for computer graphics, to address the unique challenges of representing a hyperdimensional design space. With a HyPerModel representation, visualization of the design space or design subspaces such as the Pareto subspace is possible. HyPerModels support design space analysis for adaptive sequential sampling algorithms, to detect robust design space regions or for fault detection by comparing multiple HyPerModels obtained from the same system. Significantly, HyPerModels uniquely allow multi-start optimization algorithms to locate the global surrogate model optimum in finite time. HyPerMaps defines the necessary algorithms to adaptively sample a design space, construct a HyPerModel and to use a HyPerModel for visualization, analysis or optimization. With HyPerMaps, an engineering designer has a window into the hyperdimensional design space, allowing the designer to explore the design space for undiscovered design variable combinations with superior performance capabilities.

The main problem with Turner's proposed hyperdimensional performance models is that he presumes that all the models are incomplete and inaccurate. Surrogate models are used to approximate the design space and using them to find the global optimum solution is not a logical conclusion. What we propose in this survey is the qualitative practical guidance for industrial designers who are looking for good enough solutions. Also, in Turner's study, hyperdimensional performance models are used to create robust solutions again is not possible while he is looking for single solutions on the boundary and assume that models are complete and accurate. So, in our review, surrogate models are defined and critically evaluated as

methods to approximate the reality to find the solutions which are relatively insensitive to the variations.

3.4.9. Multiple surrogates

Frequently, an individual surrogate is chosen to model a specific problem and develop an approximation function based on prior knowledge from past experience. The idea of generating an ensemble of surrogates has been used for a long time. Multiple surrogates have been used to overcome the drawbacks of other approaches. For example, adaptive sampling algorithms that add one point per cycle are readily available. These algorithms use uncertainty estimators to guide the selection of the next sampling point(s). The addition of one point at a time may be inefficient if running simulations in parallel is possible. To address this problem, (Viana et al., 2010a) proposed an algorithm for adding several points per optimization cycle based on the simultaneous use of multiple surrogates. In another study, (Song et al., 2018) analyzed the effectiveness of multiple surrogates in providing accuracy, robustness, and efficiency requirements for many specific problems. Additionally, the performance estimates and the simultaneous application of multiple surrogates are studied in . Furthermore, in the context of surrogate endpoints, multiple surrogates are used and two simple, complementary ways to address the relative benefits of multiple surrogates over a single surrogate are proposed in (Xu and Zeger, 2001). The disadvantages of composite and multiple surrogates and their lack of quasi-concavity are studied in (Karwan and Rardin, 1980). The effectiveness and found that even though empirical evidence has suggested that the Dempster–Shafer theory (DST)-based mixed surrogate approach and weighted average surrogates (WAS)-based ensemble methods

are better than individual surrogates, these methods do not generalize well to all classes of problems are debated in (Babaei and Pan, 2016).

Multiple surrogates are also extensively used in surrogate-assisted optimization. For example, (Viana et al., 2013) used the multiple surrogate efficient global optimization (MSEGO) algorithm to overcome the problem of adding several points per optimization cycle. The existing global optimization algorithms have been shown to have the potential to be adapted to locate multiple candidate designs, but the key to efficiency lies in the parallelization of optimization processes (Chaudhuri and Haftka, 2014; Villanueva et al., 2013). In addition, the trade-offs among thrust, current or power consumed, and efficiency by analyzing predictions of Pareto fronts generated from multiple surrogates are studied in (Chaudhuri et al., 2014). (Acar, 2015) argued that MAXE is more important than RMSE; in ensemble, the weight factors need to be identified by minimizing the maximum absolute cross-validation error (MAXE-CV). Multiple surrogates to reduce uncertainty in optimal point search in a surrogate-assisted optimization process are used in (Adhav et al., 2015).

The design optimization and reported the flow parameter optimization of a bi-directional impulse turbine are used in (Badhurshah and Samad, 2015). MSEGO is used to improve the convergence ratio of an uncertainty estimator called expected improvement (EI) and other modeling methods (Wang et al., 2016). Moreover, in some studies, design space reduction has been incorporated into the optimization method using an ensemble of surrogates (Song et al., 2018). Multiple surrogates are used to design a thin-walled compliant mechanism component.

The proposed optimization problem is addressed with a multiple-objective genetic algorithm in sequentially updated surrogate models.

In another engineering design problem, a sophisticated design space is explored when the shape optimization of a bluff body is performed to facilitate mixing while minimizing the total pressure loss (Mack et al., 2005). In another study, a sequential quadratic programming is used to search for the optimal point from the constructed surrogates in shape optimization of a turbomachinery blade (Samad et al., 2006). For a single problem, the choice of test surrogate could depend on the experimental design (Goel et al., 2007) while for a multiple set problems and ensemble of model with multiple regional optimized weight factors (EM-MROWF) can be useful (Yin et al., 2018) where the design space is divided into multiple subdomains, each of which is assigned a set of optimized weight factors.

Some studies analyze the surrogate model-based design optimization concern of the modeling fidelity of the approximation functions. (Bellary and Samad, 2017) address this issue using multiple surrogates based on the same data to offer approximations from an alternative modeling perspective. These authors proposed an approach to optimize the performance of a centrifugal impeller as a case study. A multiple-surrogate-assisted optimization approach is used and evaluated at various levels of fidelity. Some researchers have focused on multiple surrogates of only one surrogate type (Habib et al., 2017). For example, an ensemble of radial basis function (ERBF) method is presented to determine weights by solving a quadratic programming subproblem (Shi et al., 2016). implemented multiple-surrogate-assisted multiple-objective optimization for computationally expensive engineering design.

Many research studies have been conducted using multiple surrogates as evolutionary computation methods. These authors studied the search performance of using different metamodeling techniques, ensembles, and multiple-surrogates in a surrogate-assisted memetic algorithm (Lim et al., 2007). A multiple-surrogate model is further applied to optimize a dimpled channel and integrate multiple objectives related to heat-transfer and friction-loss into a single objective (Samad et al., 2007). Multiple surrogates are used coupled with an evolutionary genetic algorithm to identify the pareto optimal fronts (POFs) of two centrifugal pumps with different specifications to enhance their performance (Bellary et al., 2016).

The authors found that WAS performs better in both objectives than does any other individual surrogate. Multiple surrogates are used along with NSGA-II for the optimization of a `structure/control' simultaneous design approach to introduce frequency-dependent weighting functions to the H1-control synthesis framework as design variables (Bhat et al., 2010). The authors applied this method for vibration-attenuation in hypersonic vehicles. Additionally, multiple surrogate models are used along with an evolutionary algorithm that operated in parallel to combine their features and solve a costly multiple-objective optimization problem (Arias-Montano et al., 2012a).

Different metamodels are used to treat the problem of response approximation based on different theories (Basudhar, 2012). While these metamodels differed in the manner in which they addressed the approximation problem, they also had certain inherent similarities. A multiple-objective evolutionary algorithm is embedded in multiple adaptive spatially distributed surrogates of multiple types (Bhattacharjee et al., 2017). In another study, the same

group used multiple surrogates to exploit the best features of several strategies and specifically compared two possible versions of pre-selection in multiple-objective optimization (Bhattacharjee et al., 2016). A multiple-objective evolutionary algorithm with multiple adaptive spatially distributed surrogates is introduced. The multiple-surrogate-based PSO (MSPSO) framework is used, which consists of inner and outer loop optimization (Lv et al., 2018).

Subsequently, authors have attempted to use different applications to show the effectiveness of using multiple surrogates in different problems with different characteristics. For instance, multiple-surrogate modeling is used to minimize the RMS error in metamodeling (Viana and Haftka, 2008). These authors explored the use of the best PRESS solution or a weighted surrogate if a single surrogate is needed. Then, the advantages of using multiple surrogates for approximation and reduction of helicopter vibration is studied in (Glaz et al., 2009). A bi-directional impulse turbine used in a wave energy device is simulated using a CFD technique, and shape optimization is performed with a multiple-surrogate-assisted multiple-objective evolutionary algorithm in (Ezhilsabareesh et al., 2018).

Cross-validation has been used extensively to assign weights to different surrogates by generating multiple surrogates and facilitate the measurement of accuracy during the process of using multiple surrogates. However, some authors have used cross-validation to estimate the required safety margin with a given desired level of conservativeness (percentage of safe predictions) (Viana et al., 2009). These authors also determine how well they could minimize the losses in accuracy associated with a conservative predictor by selecting among alternate

surrogates. Additionally, Viana and co-authors discussed whether to use the best PRESS solution or a weighted surrogate if a single surrogate is needed (Viana and Haftka, 2008). These authors proposed the minimization of the integrated square error as a way to compute the weights of the weighted average surrogate model. The authors found that generating a broad set of different surrogates and then using PRESS as a criterion for selection are beneficial.

3.5 Model Fitting in Surrogate Modeling

To fit a surrogate model to the dataset under study, some fitting indicators are required. There are several model fitting indicators that most important ones are studied in this section. For example, in Section 3.5.1., least squares analysis is explained as the most important model fitting technique while weighted least square regression technique is discussed in Section 3.5.2. Also, in Section 3.5.3., a logarithm type of likelihood function is explained as the log-likelihood technique. Backpropagation, R-square, mean absolute percentage error, mean absolute error, and cross-validation are described in the subsequent sections.

3.5.1. Least squares analysis to create the surrogate model

The least squares (LS) analysis is a standard method in regression analyses that predicts a set of functions in an overdetermined system[1]. The objective is to identify the values of the model's parameters that fit best. Thus, the aim is to minimize the integration of the estimation errors' squares (Vinzi et al., 2010).

[1] Sets of equations that have more equations than unknowns

$$Minimize\ S = \sum_{i=1}^{n} e_i^2 \qquad \text{Equation 3.9}$$

An error is the subtraction of the response's real value from its estimation. Each data point has one error, and the eventual goal is to ensure that both the sum and the mean of the errors equal zero. In the surrogate modeling process, the LS method predicts the parameters by polynomials. Then, the fitted RSM is implemented. Subsequently, the approximated model should be almost equal to the exact model if it is a sufficient prediction of the actual model (Montgomery, 2017).

3.5.2. Weighted least square regression (WLSR)

The typical LS analysis presumes homoscedasticity, suggesting that the variance in the errors is constant. If the errors have no static variation in typical LS (this is also called heteroscedasticity), the method of WLSR is useful (Hansen et al., 2012). In the surrogate modeling process, a WLSR at a given point involves identifying the best multivariate polynomial approximation (Bettebghor et al., 2011).

3.5.3. Log-likelihood

Log-likelihood is a logarithm type of likelihood function. The likelihood is a function of combinations of coefficients and parameters in a model that creates a statistical collection of data. The likelihood is a critical tool in statistical analyses used to estimate an unknown parameter by using a data set. Usually, using log-likelihood as the likelihood is more straightforward, and this logarithm grows gradually and achieves the same value as the original

function at the optimal point. Therefore, using log-likelihood instead of likelihood is possible in searching for the maximum value of a model (Montgomery, 2017).

3.5.4. Backpropagation

Backpropagation is a technique used to compute the loss function's gradient in an ANN's weights. We use backpropagation as a method to enhance the efficacy of ANN by adapting the weights. **B**ackpropagation is also used for training the data (Demuth et al., 2014). The backpropagation process is an iterative two-step process, that is, propagation and weight adjustment. Once a given vector enters a space, it is propagated throughout the process until it arrives at the final layer. Then, using a loss function, the outcome is analyzed against the ideal result, and we can compute the error value of each neuron in the last layer. Accordingly, starting from the output, the values of the error are propagated backward, and this cycle continues until each neuron attains a corresponding error value that approximately denotes its contribution to the primary output (Chauvin and Rumelhart, 2013).

3.5.5. R-square (R^2)

R-square is a percentage of the deviation of the response variable and is estimated using the data of the factors. In surrogate models, we use R-square to examine the level of the model's accuracy. The adjusted R^2 is another version of $R^{2,}$ that considers the number of parameters, R^2 and adjusted R^2 are some of the leading indicators of accuracy in simulation-based design. However, R-square and adjusted R-square can be inadequate, and a high R^2 can be elusive; thus, using additional different data points is necessary (Simpson et al., 1997). Several other

measures, such as root mean square (RMS) and mean absolute error (MAE), are used for examining the model accuracy (Venter et al., 1997).

3.5.6. Mean absolute percentage error (MAPE)

This indicator is very frequently used for testing the accuracy of the approximation because it is interpretable and scale-independent (Dobler and Anderson-Cook, 2005). However, MAPE may generate undefined or infinite values, when the real values are insignificant (near to 0). Thus, in many disciplines, MAPE can be problematic because having minimal values is a standard issue in many experiments (Makridakis et al., 1998). To address this issue in MAPE, another version of this measure was developed to measure the accuracy of the estimation, which is called MAAPE[2] (Kim and Kim, 2016).

3.5.7. Mean absolute error (MAE)

MAE is another indicator used to measure two variables' variance. For example, if A and B are variables in two different experiments of one clear case, the values of A and B are comparisons between the estimated and observed phenomena (or between current time and previous time). Thus, if we generate a scatter chart (such as the chart shown in Figure 3.2) of m points, where the point k has a location (a_k, b_k). (See Equation 3.10)

$$MAE = \frac{\sum_{k=1}^{m} |b_k - a_k|}{m} \qquad \text{Equation 3.10}$$

The MAE is the mean distance (vertical) from each point to the line A=B. MAE is also the average distance (horizontal) between each point and the B=A line.

[2] Mean arctangent absolute percentage error

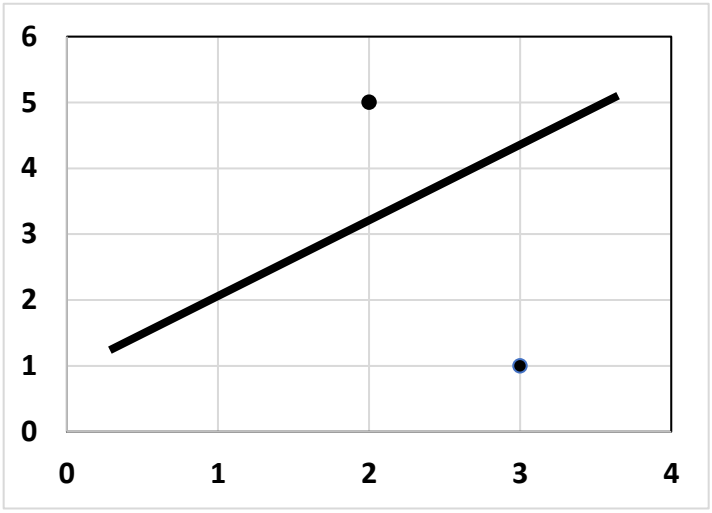


Figure 3.2. Comparison between two experiments in an engineering design problem.

3.5.8. Cross-validation

This validation method is used to evaluate the validation of the outputs of a statistical study and generalize the results to other sets of data. In this method the data is divided into test and training data sets and the process is repeated several times (Geisser, 1993). In surrogate modeling, if we wish to fit a model to a given known data set, we train the model by running it with the training data set. Then, the surrogate model is validated against the test data. Using this method gain insight into whether the surrogate model can be generalized into a new independent set of data (Lucas, 1994).

Each replication of the cross-validation technique contains three phases; in phase one, we divide the data into the two subsets, apply the analysis to the training subset and validate the study using the testing subset. Many replications are performed to avoid variability with distinct partitions of the original data, and the mean of the results over the replications is used as the final validation outcome (Lucas, 1994).

We use this method instead of conventional methods for validation, such as dividing the set of data into 60 percent for training and 40 percent for testing. The lack of adequate data to partition into two training-testing data sets can lead to a loss of significant modeling capability and testing efficiency to partition into two training-testing datasets (Seni and Elder, 2010).

One of the greatest difficulties faced by designers is a situation in which two computer codes are available, and one code is more accurate than the other but also more expensive (time consuming) to run. The issue of building surrogate models based on combining the high-accuracy fitting method and low-accuracy fitting method has drawn significant attention. Related work includes Kennedy and O'Hagan (Kennedy and O'Hagan, 2000, 2001), Higdon et al. (Higdon et al., 2004), Reese et al. (Reese et al., 2004), and Qian and Wu (Qian et al., 2008). The fundamental idea of these methods is to run the high-accuracy and low-accuracy data with a pair of nested space-filling designs (Qian et al., 2009) and then fit a surrogate model based on the low-accuracy data and refine it by incorporating the more accurate high-accuracy data. These methods work in a uni-stage experimentation fashion as the run sizes of the high-accuracy and low-accuracy data need to be predetermined, and all data from the two sources are obtained in a single stage. Additionally, Xiong and Wu (Xiong et al., 2013) propose a sequential framework for designing and analyzing a pair of high-accuracy and low-accuracy data. This framework is desirable in situations in which it is difficult to predetermine the number of runs needed for the high-accuracy and low-accuracy data. This framework is also useful when the high-accuracy and low-accuracy methods cannot be performed in a large number of runs simultaneously due to experimental constraints. In another study, (Zhou et al.,

2011) proposed a new approach to the emulation of computer code with qualitative and quantitative factors. The approach inherits the flexibility of the unrestrictive correlation structure of qualitative factors used by Qian, et al. (Qian et al., 2008) but replaces their complicated estimation procedure with a clever parameterization using hypersphere decomposition, which was initially proposed by Jackel and Rebonato (Jaeckel and Rebonato, 1999) for modeling correlations in financial models. This new parameterization essentially turns the required optimization problems with positive definite constraints proposed by Qian et al. (Qian et al., 2008) into standard nonlinear optimization problems with box constraints.

3.6 Comparison of Chosen Surrogate Models

The appropriateness of a surrogate model depends on the trade-off between the model accuracy and the computational expense of its development and execution, which we consider the computational time. Because no surrogate model is perfect, any surrogate model used for a system's physical behavior can be refined further to increase its accuracy, although, unsurprisingly, at an increased computational expense (Paiva et al., 2010). On the other hand, an alternative approach to reduce the computational expense is to replace detailed simulations with simplified approximate simulations, thereby sacrificing accuracy for reduced computational time. In this case, a strategy to maintain accuracy along with reducing the computation time can be to integrate data from approximate and detailed simulations to build a surrogate model that describes the relationship between output and input parameters (Qian et al., 2006). Therefore, the question asked by a designer is, ‘How much refinement of a surrogate model is appropriate for a particular design problem?' Value-of-information has been addressed

in the engineering design literature to determine whether to make a decision using the available information or collect more information before reaching a decision (Messer et al., 2010). However, the main drawback of applying existing value-of-information-based metrics for model refinement problems is that the existing metrics only account for time, size and accuracy separately; these metrics do not account for all three combined in surrogate models, and this consideration has an impact on design decisions (Panchal et al., 2009).

To address the lack of current metrics in the context of model refinement, in this chapter a value-of-information-based method for identifying the appropriate amount of refinement of surrogate models is proposed. The approach introduced in this chapter can be utilized by designers and analysts in developing useful surrogate models for specific design problems while efficiently utilizing their model development resources. In this subsection, we compare the chosen surrogate models by balancing accuracy and computational time. We consider the dimensionality of the problem (the number of factors or design variables), the surrogate modeling process time and the level of precision in this comparison.

3.7 Results of Critical Evaluation of Literature on Surrogate Modeling

In this section, we outline the primary applications of each DOE, selecting a model to represent the data and model-fitting approaches to evaluate the fitness of the chosen models. We first discuss how we review the surrogate modeling process, and then, we recommend more appropriate approaches based on evaluation metrics, computational time and accuracy.

3.7.1. Review of the applications of DOE, data representation models, and model-fitting methods

In this subsection, we summarize the different DOE techniques, various models used for representing data and methods used to evaluate the fitness of each model to the generated data by each DOE method. Here, we also discuss several surrogate modeling methodologies. For example, we conclude that if we use the fractional factorial design as a DOE and a linear or quadratic polynomial as the model representing the data, which are generated by a fractional factorial design, we can use LS regression as the model-fitting method; this process is a type of response surface methodology (RSM). An initial summary of different surrogate modeling procedures is provided in Table 3.3 (Simpson et al., 2001).

Table 3.3. Surrogate modeling techniques.

Design of Experiment	Surrogate Model Choice	Model Fitting
Classic Methods	▪ Multivariate Adaptive Regression Splines (MARS)	▪ Log-likelihood
▪ (Fractional) Factorial	▪ Polynomial (Linear, quadratic, or higher)	▪ Best Linear Unbiased Predictor (BLUP)
▪ Alphabetical Optimal	▪ Kriging	▪ (Weighted) Least Squares Regression
▪ Central Composite	▪ Splines (Linear, cubic, or NURBS)	▪ Best Linear Predictor
▪ Box-Behnken	▪ Radial Basis Functions (RBF)	▪ Sequential or Adaptive Metamodeling
▪ Plackett-Burman	▪ Artificial Neural Network (ANN)	▪ Multipoint Approximation (MPA)

Space-filling Methods ▪ Hammersley Sequence ▪ Simple Grids ▪ Minimax and Maximin ▪ Orthogonal Arrays ▪ Uniform Designs ▪ Latin Hypercube Hybrid Methods Importance Sampling Discriminative Sampling Directional Simulation Sequential or Adaptive Methods	▪ Least Interpolating Polynomials ▪ Support Vector Machine (SVM) ▪ Knowledge Base or Decision Tree ▪ Hybrid Models ▪ Gaussian Process	▪ Back Propagation (for ANN) ▪ Entropy (inf.-theoretic, for inductive learning on decision tree)

Random or Human Selection

3.7.2. Suggestions for DOEs, model selection and application, and model-fitting methods

In this subsection, we develop scenarios that engineering designers may face while addressing problems. The computing time, number of design variables, and desired accuracy level for achieving an efficient surrogate modeling process are the main critical factors that should be considered. Additionally, we consider accuracy in the explanation of the following nine different states: a-i, a-ii, a-iii, b-i, b-ii, b-iii, c-i, c-ii, and c-iii. Then, we explain the time-size-accuracy triangle (see Figure 3.3) in more detail.

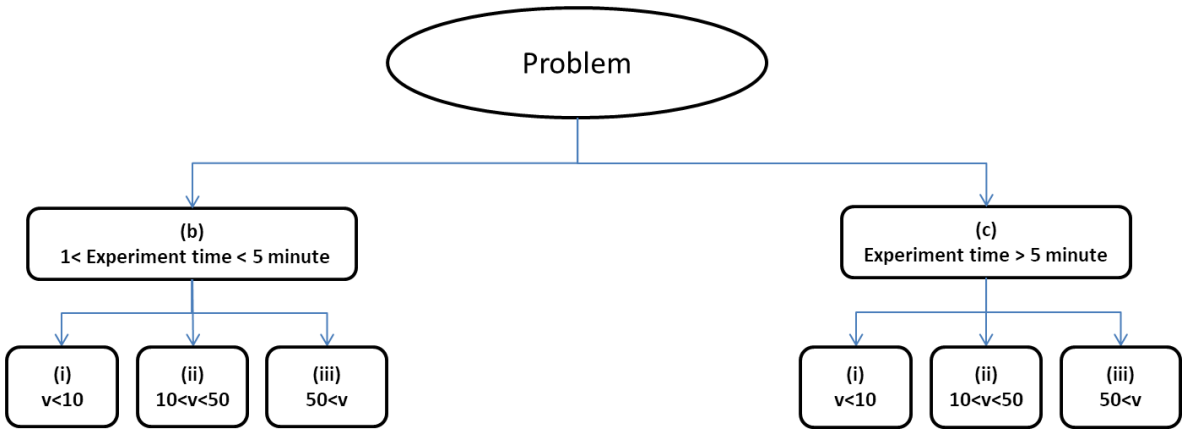


Figure 3.3. Problem characterization based on the two aspects of time and size.

Note: v indicates the number of variables.

As shown in

, we always should balance “the needed speed of a surrogate modeling process," "the required accuracy" and "the number of dimensions or degree of complexity we aim to maintain within the model." To achieve this trade-off, information about the available tools and their

characteristics, capabilities, pros, and cons and problem-specific knowledge are needed; we provide the reader with this knowledge.

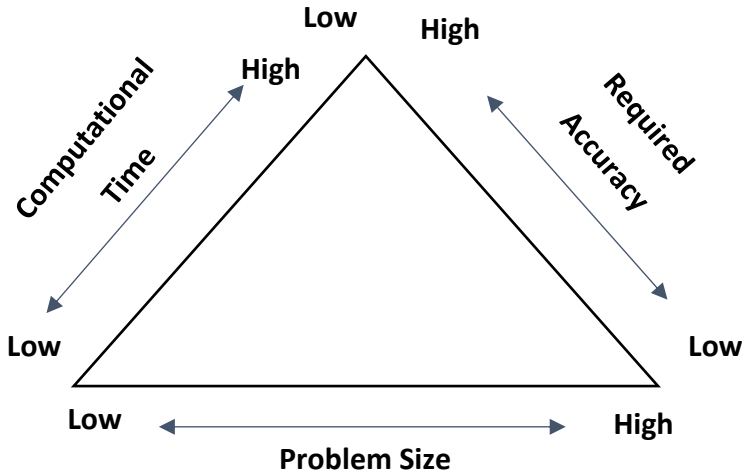


Figure 3.4. Time-size-accuracy triangle

First, we review some state-of-the-art approaches to addressing the time-size-accuracy triangle. Deb et al. in (Deb et al., 2017) classified the various metamodeling methodologies for multiple-objective optimization and proposes a comparative analysis by explaining the pros and cons of each method. Another primary outcome of this chapter is the classification of some surrogate modeling methods into six categories.

In addition, Deb et al. in (Tutum and Deb, 2015) and (Deb et al., 2017) applied two distinct selecting functions based on the following two recent ideas: (i) the KKT proximity measure function and (ii) the multimodal-based evolutionary multiple-objective (MEMO) selection function. Subsequently, they apply outcome surrogate modeling approaches to several standard two- and three-objective constrained and unconstrained test problems; the authors compared the efficiency of two surrogate modeling approaches, that is, ANN and Kriging, by applying

these selection function methods to these problems. According to their results, the performance of the MEMO-based approach is high, and the ANN surrogate modeling technique is generally better able to approximate the multimodal selection function landscape portrayed by the overall approach. Additionally, the MEMO selection function enables the process to be completed using only a fraction of the solution evaluations (limited to 500 to 2000) compared to the hundreds of thousands of solution assessments.

Therefore, Deb and co-authors conclude that the MEMO method and ANN are **more** appropriate if the goal is to reduce the number of runs and, accordingly, the required time of the surrogate modeling process. Based on their recommendation, the use of different architectures and deeper ANNs for better accuracy models, the use of other selection function modeling after successful EMO approaches and extending the approach to many-objective problems need to be further studied and are suggested as future research topics.

The authors also increased the understanding of the behavior of EMO approaches when confronting a substantial number of objectives and proposed some initial empirical analyses for multiple problems with multiple objectives (Li et al., 2017). The authors found that the reference-point-based Nondominated Sorting Genetic Algorithm II (R-NSGA-II), which is suggested when searching for a region of interest (ROI)s, results in very competitive and consistent solutions for addressing problems with many objectives.

In addition to the extensive work investigating massive objectives and constraints, in (Deb and Myburgh, 2016), the authors proposed a method to simultaneously meet the time and size requirements. Evolutionary Algorithms (EAs) can meet the size requirements suitably, but the

run time is not efficient using these methods. However, in this study, Deb et al. proposed a method that can identify the optimum solution very quickly with 300 to 2000 integer variables. Additionally, the method can manage computational complexity by addressing 50000 to 10^9 variables. It is very promising that the authors can solve such a large problem in seconds.

A cross-validation method is propose for creating more accurate surrogates (Viana et al., 2010b). The authors found that cross-validation is useful for surrogate selection, although it is inaccurate when there are few data points (Viana et al., 2010b). The authors concluded that variable screening is a useful process to reduce the price of the metamodeling process and that extreme dimensionality reductions are possible. These authors also recommended other methods, such as non-depersonalization or PCA, to improve the accuracy of the approximation. Sequential sampling also enables us to use a limited computational budget efficiently (Viana et al., 2013).

The multiple-surrogate method is more appropriate (or based on the authors' results better-performed) than Kriging if we use more than 100 experimental designs. This method is more appropriate if we wish to obtain the results within 5 minutes. Additionally, the authors apply the proposed method to problems with less than ten variables; thus, this approach is appropriate for the (b-i) type problems in our taxonomy shown in Figure 3.5.

Furthermore, the comprehensive literature review shown in Table 3.4 confirms that our results regarding the classification of methods based on time-size-accuracy triangle criteria are justifiable.

Table 3.4. Critical literature review table: survey of engineering application of DOE, data representation models, and model fitting.

Papers/Criteria	Computational Time		Number of Variables			Method and Accuracy Level
	1<time<5 minute	time>5 minute	v<10	10<v<100	100<v	
(Viana et al., 2010b)	✓		✓			RSM - Medium
(Koch et al., 1999)	✓			✓		Kriging - High
(Li et al., 2017)		✓			✓	Neural Network – Medium
(Li et al., 2017)	✓				✓	Genetic Algorithm II (R-NSGA-II)
(Jiang et al., 2016; Panchal et al., 2008)	✓			✓		Kriging - High
(Yang et al., 2006)		✓			✓	Neural Network – Medium
(Geisser, 1993; Jiang et al., 2016; Jin et al., 2001; Viana et al., 2010b)	✓		✓			RSM - Medium
(Maier and Dandy, 2000)		✓			✓	Neural Network – Medium

Papers/Criteria	Computational Time		Number of Variables			Method and Accuracy Level
	1<time<5 minute	time>5 minute	v<10	10<v<100	100<v	
(Myers et al., 2016)	✓		✓			RSM - High
(Schmidhuber, 2015; Seni and Elder, 2010)	✓				✓	GE – High
(Krauss et al., 2017)	✓				✓	Neural Network – High/ GE - High

After reviewing the surrogate modeling literature, the results of this analysis using the proposed taxonomy are summarized in Figure 3.5.

- *b-i) Experimental time is less than 5 minutes and the number of variables is less than 10*

When the desired experimental time is less than 5 minutes and the number of variables is less than 10, response surface methodology performs well; however, this method is not appropriate if the functions are nonlinear, or we have discrete and binary variables and continuous variables. Under such circumstances, we can use Kriging, but we should consider that utilizing this method leads to more complexity and that we can only use this method for deterministic applications.

- *b-ii) computational time more than 5 minutes and no more than one variable*

If the desired experimental time is between 5 and 100 minutes and the number of variables is between 10 and 100, Kriging is a more appropriate method, and the response surface model

loses efficiency. In addition, we can use ANN instead of RSM; however, if we need more accuracy, Kriging is the better choice. Meanwhile, we can use variable screening and other dimensionality reduction methods if the number of factors is greater than 10 (Koch et al., 1999).

- *b-iii) The number of variables is very high, greater than 100*

If the number of variables is very high, greater than 100, response surface models are not useful, and we can use neural networks, genetic algorithms, and other evolutionary algorithms. Additionally, we can use Kriging with some adjustments. However, using Kriging or RSM is possible if we perform some preliminary dimensionality reduction steps using variable screening and PCA.

In this category, concerning time, extensive research studies have been performed (Deb et al., 2017; Deb and Myburgh, 2016; Li et al., 2017) to demonstrate that evolutionary algorithms can execute the process quickly. However, these algorithms are not as accurate as they are fast, and sometimes, we need to integrate the algorithms with other more precise methods, such as Kriging (Viana et al., 2013). Multiple surrogates represent a new approach to addressing such situations in which we can combine more accurate surrogates, such as those obtained by Kriging, with faster surrogates, such as every evolutionary method, to compensate for the weakness of each surrogate with the advantages of other surrogates.

- *c-i) The computational time can be more than 5 minutes*

In this category, if the computational time can be more than 5 minutes and we do not have more than one variable, we can use RSM. However, the time can be reduced using Kriging or by using evolutionary algorithms, such as genetic algorithms and neural networks.

- *c-ii) The experiment time is less than 5 minutes, and the number of variables is between 10 and 100*

If we have more experiments which require than 5 minutes and the number of variables is high, for example between 10 and 100, we can use Kriging (for high accuracy) and every evolutionary algorithm (for a more rapid process). In addition, we can use RSM after implementing some dimensionality reduction methods.

- *c-iii) The number of variables is more than 100*

Evolutionary algorithms, such as ANN and genetic algorithms, are always appropriate if the number of variables is immense. Therefore, in this category of problems, it is better to use evolutionary algorithms. Albeit Kriging can be used if we use some dimensionality reduction methods before the process of surrogate modeling.

This six-type classification, which contains a wide range of surrogate models, demonstrates the differences among the models and allows us to adopt more appropriate model for a specified variable size and level of accuracy in a given amount of time. In Table 3.5, we summarize the general characteristics of the surrogate models. Based on our survey of the literature and the results reported in multiple studies, various methods are recommended for various situations in Figure 3.5. For example, Kriging is a method that results in high levels of accuracy and adequately performs for large problem sizes but not for problems with more than approximately 50 variables. Additionally, some authors mention in their outcomes that due to the complexity of the modeling, more time is needed to run Kriging models (Jin et al., 2001).

Table 3.5. General characteristics of surrogate models

Model selection	**Time**	**Dimensionality**	**Accuracy**	**Complexity**	**Linearity**	**Deterministic/ Stochastic**
Response Surfaces	Rapid if can be used	Useful if the number of factors is less than 10	Not very high but acceptable	Easy to apply and well-established	More appropriate for linear functions	Appropriate if errors are random
Kriging	Slow but accurate	Applicable if the number of factors is more than 50	Very high	Complex but very flexible	Can be implemented for both linear and nonlinear functions	Appropriate for deterministic uses
Evolutionary Algorithms, such as GA, ANN	Quick but not very accurate	Appropriate for problems with very many parameters (~10,000)	Low level of accuracy	Useful if the complexity is very high	Proper for very nonlinear problems	Best performed in deterministic uses

Thus, we placed the Kriging models in the triangle relatively close to the desired accuracy and relatively high or medium problem size, which needs much more computational time than other methods. Kriging is placed close to the desired accuracy because it has a high level of accuracy and is at a medium distance from the problem size vertex because it is good enough for high-dimensional problems, but we placed it far from the computational time vertex because it is time-consuming.

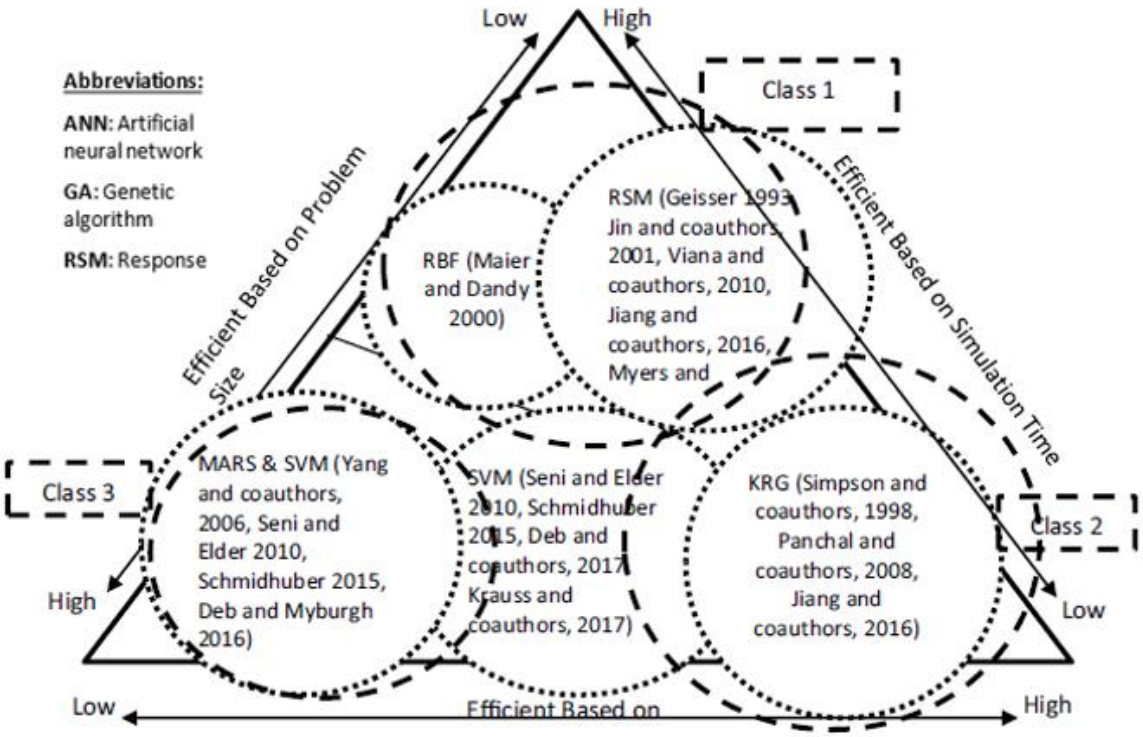


Figure 3.5. The Classification of some surrogate modeling techniques in a time-size-accuracy triangle.

Another example is ANN, which performs well for high-dimensional problems with high levels of nonlinearity. ANN has also been developed in recent years, and although the initial versions were poor at the computational time criteria (they were very time-consuming), they are currently very fast. However, ANN is not very accurate as discussed in the literature (Broomhead and Lowe, 1988; Cheng and Titterington, 1994; Demuth et al., 2014) literature. RSM has been used for a long time with high efficiency for small-sized problems with acceptable levels of accuracy, but it is not very rapid for even some small problems. RSM loses its efficiency as the size of the problem grows, and if the functions are nonlinear (Khuri and Mukhopadhyay, 2010; Myers et al., 2016; Simpson et al., 1998; Simpson et al., 1997; Venter et al., 1997). Therefore, we placed RSM close to the accuracy and computational time vertexes because it is better in these criteria than in the size criteria. RSM is a very convenient and easy

to use surrogate modeling method, and due to its characteristics, even though it is not very fast, accurate or efficient in high-dimensional problems, RSM is frequently used.

Finally, we studied genetic algorithms in the literature and found that this evolutionary algorithm (EA), similar to other types of EAs, is very fast in modeling and better in solving the models, but its most powerful aspect is its ability to address high-dimensional problems (Lin et al., 2015). Additionally, although it was not accurate in the past, similar to other EAs, some recent studies (Cho et al., 2014; Deb et al., 2017; Deb and Myburgh, 2016; Yang et al., 2006) have shown that the updated versions have significantly improved in accuracy over time. Thus, we decided to place this method at the center of the triangle but somewhat close to the size and time criteria.

As the results have shown, conventional methods, such as RSM and Kriging, are generally more appropriate if we wish to achieve more accurate surrogate models, and EAs are more appropriate if we wish to address high dimensionality and high linearity and obtain faster results. However, in this chapter, we find that EAs have achieved higher accuracy than they did previously; specifically, GA is currently faster and more accurate. In the future, studying deep neural networks and comparing these networks to genetic algorithms could be an exciting research topic. The idea of using multiple surrogates, as proposed in (Viana et al., 2013), enables EAs to be integrated and compensate for the weaknesses of various surrogate methods; thus, this concept is an excellent area for future research. Finally, the time of the simulation is strictly associated with the circumstances under which the models run, e.g., the power of the computer used. Therefore, these circumstances should always be considered in comparisons.

3.8 Verification and Validation of Critical Evaluation of Literature on Surrogate Modeling

Verification and validation were addressed from the perspective of dissertation chapters in Chapter 1. In the context of surrogate modelling and predictive analytics, verification and validation consists of the following activities (Panchal et al., 2013):

1. Individual Model verification and validation – a single model focusing on single length and/or time scales.

2. Multiscale Model verification and validation – single model or coupled set of models spanning multiple length and/or time scales in an integrated manner.

3. Multi-physics Model verification and validation – ensuring the mathematical and physical consistency of modelling framework spanning multiple phenomena.

4. Design Process verification and validation – ensuring that the design process in its configured form yields a solution that meets design requirements.

5. Design (outcome) verification and validation – comparing design outcomes to system-level requirements.

3.8.1. Individual model verification and validation

Model verification and validation has received significant attention in the past years due to advent in simulation-based design technologies. The following tasks are associated with model verification and validation (Sargent, 2010); the process of verification and validation is illustrated in Figure 3.6.

• Conceptual model verification and validation: process of validating whether the assumptions underlying a model and its sub-models are correct and that the representation of the system including models and sub-models are correct and reasonable for the intended study.

• Model verification: Process of assuring that the computer model is "good enough" in terms of accuracy of representation of a conceptual model.

• Operational validation: Process of determining whether the computerized model is sufficiently accurate for the needs of the simulation study.

• Data validation: Checking the accuracy and consistency of the numerical data used to support the models in the simulation study.

Multiscale model verification and validation is important as valid individual models for specific length and time scales won't necessarily result in valid multiscale models across scales.

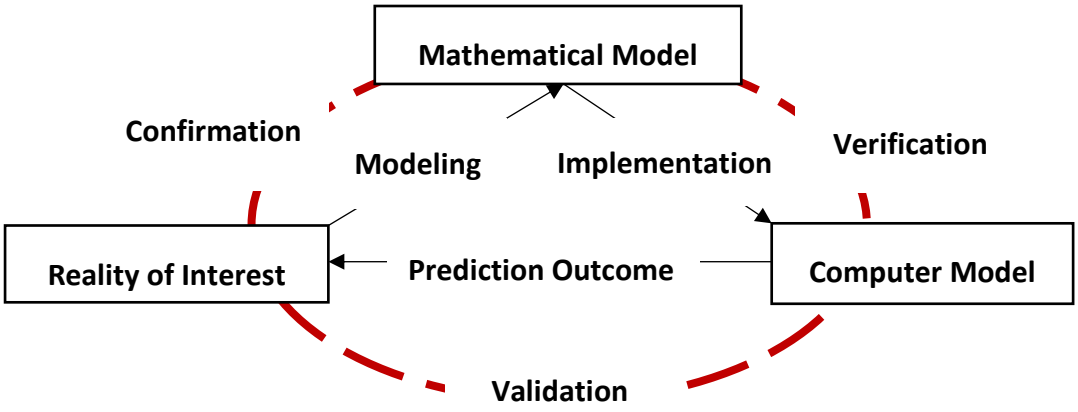


Figure 3.6. Model verification and validation process (Sargent, 2010).

3.8.2. Multiscale model verification and validation

The following tasks are involved with multiscale model verification and validation:

• Compatibility validity: Compatibility validity is the process of determining whether the input ranges of an upper-level model is consistent with the domain of outputs of a lower-level model. This ensures whether the output domain of the lower-level model is a subset of the valid input domain of the upper-level model.

• Uncertainty propagation check: The goal here is to check that the effects of uncertainty at lower length scales do not amplify beyond the desired uncertainty bounds or limits set for which the design decisions are to be made. This can be viewed both from bottom-up and top-down perspectives. From a top-down perspective the uncertainty limits allowable for a system is used to determine the allowable uncertainty limits for lower scales and thereby manage the propagation across a chain of models.

3.8.3. Design process verification and validation

The goal in design process verification and validation is to ensure that the design process yields design solutions worthy of investigation satisfying the design requirements. In the simulation-based design of complex systems, design processes represent the way design decision networks and simulation models are configured to achieve the design task. One approach to verify and validate a design process is with the help of the verification and validation square framework with a square consists of four quadrants, as shown in Figure 3.7.

1. Theoretical Structural Validity: Is the design method internally consistent?

Internal consistency of the design method is checked – this includes, checking the logical Internal consistency of the design method is checked – this includes, checking the logical soundness of the constructs used in the design method both individually and integrated.

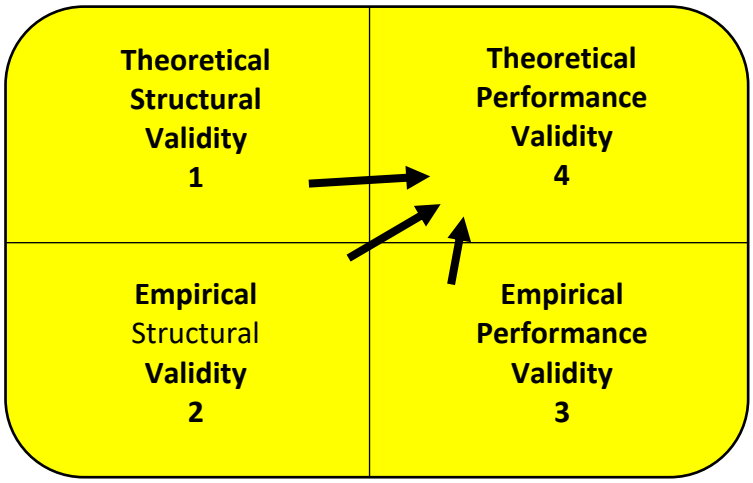


Figure 3.7. The verification and validation square (Pedersen et al., 2000; Seepersad et al., 2006).

2. Empirical Structural Validity: Are the example problems appropriately chosen for testing the design method? The appropriateness of the chosen example problems to test the efficacy of the design method is checked.

3. Empirical Performance Validity: Does the application of method to the sample problems produce practical results? Checking the ability of the design method to produce useful results worthy of investigation for the chosen example problems.

4. Theoretical Performance Validity: Is the design method applicable for the other problem? Here the ability of the design method to produce useful results beyond the chosen example problems is established. This requires the designer to take a "leap of faith" which is supported by the confidence gained by carrying out verification and validation process 1 – 3 in establishing the generic nature of the design method.

3.8.4. Design outcome verification and validation

The goal here is to ensure the validity of the design outcome rather than the prediction models used for the design. The process involves gaining confidence in the resulting design when compared with the system-level design requirements. Experiments are generally carried

out to test the design outcomes. Li and co-authors propose an approach for design outcome validation. The approach is illustrated for a simple cantilever beam design subject to vibration.

3.8.5. Verification and validation in this dissertation

In this dissertation, the different verification and validation approaches described are used to verify and validate the design methods, simulation models, prediction models, and design results. The verification and validation square framework are used to verify and validate the surrogate model selection architecture proposed in Chapter 3 of this dissertation. The architecture is tested for the hot rod rolling process chain problem introduced in Chapter 4. Four other example problems are used for achieving this. The first example involves macrostructural design of a hot rolling process chain involving the product, which is discussed in Chapters 6 and 7 of the dissertation. The second example involves designing a blue pipe. Both these examples are discussed in Chapter 7 of the dissertation. The comprehensive example problem discussed in Chapter 8 involves the multi-echelon, multi-channel, green supply chain design where a surrogate model used to map the results of one level of the model into the other. This comprehensive example is used to test the partitioning-approximation-coordination method developed in this dissertation and serves the Theoretical and Empirical Structural and Performance validations of the design method. Individual, surrogate models and ensemble of surrogate models used in the example problems are tested in terms of concept, accuracy, operation, and data and are discussed in detail in Chapters 4, 5, 6 and 7.

In Chapter 3, Gap 1, which is on "classifying the surrogate models based on common selection criteria" is addressed through answering the Research Question 1, "What are the main classes of the design of experiment (DOE) methods, surrogate modeling methods and model-fitting methods?". We hypothesized that surrogate modelling methods can be classified based on the problem size, computational time, and accuracy. The key outcome of addressing this gap is to build

a framework to provide guidance for researchers and practitioners to choose the most appropriate surrogate model based on incomplete information about an engineering design problem.

3.9 Closing Remarks on Critical Evaluation of the Surrogate Modeling Literature

In the simulation-based realization of complex systems we are forced to address the issue of computational complexity. Surrogate models are used to replace expensive simulation models of engineering problems. We suggest that a framework for selecting a more appropriate surrogate model for a given function with specific requirements is lacking. To address this gap, we hypothesize that a trade-off among three main drivers, namely, size (how much information is necessary), accuracy (how accurate the model must be) and computational time (how much time is required for the surrogate modeling process) is needed. In the context of these hypotheses and our critical review of the state-of-the-art surrogate modeling literature we summarize our findings to three questions in Table 3.2:

Question 1: What are the main classes of design of experiment (DOE) methods, surrogate models and model-fitting methods based on the requirements of size, time and accuracy?

Finding 1: As shown in Table 3.5 and Figure 3.5 based on three critical characteristics identified through the critical evaluation of the literature, six different categories of the surrogate models are introduced**.** This classification is a framework for selecting an efficient surrogate modeling process to assist those who aim to select a more appropriate surrogate modeling technique for a given function.

Question 2: Which surrogate model is suitable based on the critical characteristics of size, time and accuracy? Which DOE is appropriate based on the requirements of size, time and accuracy?

Finding 2: As shown in Section 3.7.2, Figure 3.5, Artificial Neural Network, response surface models, and Kriging are more appropriate when the following characteristics are considered large problem size, less computation time and high accuracy, respectively.

Question 3: Which DOE is more suitable based on the critical characteristics of the requirements of size, time accuracy?

Finding 3: As shown in the Section 3.7.2, and based on the answer to the Question 2, Latin hypercube, fractional factorial design, and D-Optimal are appropriate DOEs for of large problem size, less computation time and high accuracy, respectively.

We offer a designer or decision make the opportunity of particularizing the choice of surrogate model for specific situations. However, sometimes, deciding which surrogate model to use is difficult. As shown in Figure 3.5, if only one or two priorities are to be considered, the decision of which method to use is not difficult. For instance, if computational time is the only concern, ANN and RSM are good. If accuracy and problem size are the most important issues, Kriging is the appropriate method. The central question here is which of the surrogate models or which combination of models should be used if we wish to achieve a balance among these three priorities. A possible answer is using multiple-surrogate and cross-validation approaches. In addition, what we presented here can help us develop a rule base for automating the process of surrogate model selection. This is becoming increasingly important as computational platforms help us to move closer to automating design processes. We discuss

this matter in Chapter 6. Now, that we have completed the surrogate modelling automation concept. It is time to discuss how different types of variables can be dealt with some modifications in surrogate modelling process. In the next two chapters (Chapters 4 and 5), the focus is on using surrogate modelling when we have temporal and spatial variables in the dataset.

In Chapter 3, the Gap 1, which is on classifying the surrogate models based on common selection criteria is addressed and the research question of "What are the main classes of the design of experiment (DOE) methods, surrogate modelling methods and model-fitting methods?" (Research Questions 1) is answered by posing the hypothesis of "surrogate modelling methods can be classified based on the problem size, computational time, and accuracy". This hypothesis is tested and proved to be correct, and the outcome is published in (Alizadeh et al., 2020a).

CHAPTER 4 APPROXIMATION OF REALITY USING TEMPORAL SURROGATE MODELING

The objective in this chapter is to introduce how time series models and temporal predictive models can be used as surrogate models. We incorporate the time series and particularly the concept of the lag with the surrogate modeling in this chapter. We proposed the idea of temporal surrogate models by adding this feature (time) as a new feature in the process of surrogate modeling. We also bring the idea of ensemble of surrogates by using the RF as an ensemble of decision trees and machine learning method. The findings in this chapter can be used by people who are working on surrogate models and forecasting, those who wants to incorporate forecasting methods and time series models with surrogate modeling. Also, it is of use for the practitioners who want to implement expert systems and machine learning-based forecasting methods in different engineering design applications. For example, decision makers in water and energy planning can use the models created in this chapter to predict the inflow of dams based on monthly weather conditions.

Surrogate models have been used to replace computationally expensive analysis models in engineering design problems. However, time-dependent variables and historical data are usually ignored in the surrogate modeling process. For instance, in a dam network design, using hydraulic simulations to estimate the water flow is computationally expensive, and the data is in the form of time series. So, we need time-dependent surrogate models to replace these simulations and manage this computational complexity. In this chapter, we describe surrogate models to predict the amount of water flow into a reservoir. The challenge is that the flow is a time-dependent variable, and we need to incorporate time-series into surrogate models. Thus, there are three contributions: (1) using surrogate modeling to predict flow for dam network design, (2) incorporating time series analysis in surrogate models for water network design, (3)

using an ensemble of surrogates to increase the accuracy of prediction. We also demonstrate how to integrate surrogate models and machine learning with time series analysis for more accurate and faster prediction. Due to the availability of data, we use the Buffalo Reservoir on the Red River Basin as an example. Based on the time series data for flow, evaporation, precipitation, and maximum and minimum temperature, three surrogate models are used to examine the impact of integrating time series into surrogate models.

These are multivariate autoregressive integrated moving average (MARIMA), a classic time series analysis method; artificial neural network (ANN), and random forest (RF) methods, two machine learning surrogate models. We use seven different time lags as features within an RF model, as an ensemble of surrogate models, and predict the flow for seven-time steps ahead. We successfully incorporate the time series data and particularly the concept of the time lag within surrogate models. We show that RF as the ensemble of surrogates provides more accurate predictions than the other two surrogate models. Although this method has been demonstrated for the Red River Basin, it could also be applied to designing anything in which time-dependent flow is an issue, for example, in biomedical applications, the management of manufacturing processes and product sales as well as any products in which fluid flow is an issue.

Nomenclature

ACF = Autocorrelation Function

AI = Artificial Intelligence

ANFIS = Adaptive Neuro-Fuzzy Inference System

ANN = Artificial Neural Network

ARIMA = AutoRegressive Integrated Moving Average

DT = Decision Tree

FFNNs = Feed-forward neural networks

LMBP = Levenberg–Marquart Back Propagation

MA = Moving Average

MARIMA = Multivariate AutoRegressive Integrated Moving Average

PACF = Partial Autocorrelation Functions

RF = Random Forest

RMSE = Root Mean Square

SSA = Singular Spectrum Analysis

WMRA = Wavelet Multi-Resolution Analysis

VAR = Variable Autoregressive

4.1 Frame of Reference on Building Temporal Surrogate Models

Time series analysis, a series of data points indexed (or listed or graphed) in time order, is used to study the characteristics of the target variable concerning time-independent variables. This means to estimate the response variable in the name of prediction, use the time as the point of reference. Using this method, designers can find the periodical trends in their data and understand the underlying causes of these systematic patterns over time.

They enable design engineers to foresee next-generation product features before they become insignificant or mainstream by capturing the changes in consumer preferences (as they relate to product design) over time (this is called trend mining in product design) (Tucker and Kim, 2011). An accurate and efficient time series analysis is fundamental for building digital

twins. As a critical part of a digital twin, the time series prediction of the performance and efficacy of complex equipment must be used during the lifecycle of the complex equipment, including the design, manufacturing, operation, and maintenance stages (Hu et al., 2021). Therefore, it is essential to build real-time and leading time insight, a critical technology for digital twin-based product design, service, and manufacturing (Tao et al., 2018). These methods can also be used in uncertainty estimation to predict technology evolution (Zhang et al., 2019).

Temporal and spatial analysis has gained attention from the engineering design community recently. A well-studied example is a time-dependent system reliability analysis, where there is a deterioration in the properties of the structure's material or a random process such as random loading. There are several solutions to the time-dependent reliability problem. One of the most common ones is the outcrossing rate method.

The performance functions' outcrossing rates at a random time point are computed and converted to reliability assuming the Poisson, Markov, or their enhanced versions. For instance, in time-dependent reliability problems, an outcrossing rate model and its efficient calculation approach are developed for systems design problems, and based on the presented model, a time-dependent system reliability analysis method is proposed (Jiang et al., 2017). Surrogate models are important methods used for time-dependent reliability problems to manage the non-linear extreme value functions (Hu and Du, 2015). In another study, a Kriging model is used to build a nested response surface of time corresponding to the extreme value of the limit state function (Wang and Wang, 2012).

Single-loop Kriging (SILK) surrogate modeling method for time-dependent reliability analysis are used to manage the computational complexity of a double-loop procedure (Hu and Mahadevan, 2016). Also, an envelope approach is developed to time-dependent mechanism reliability defined in a period where a specific motion output is required (Du, 2014). In another study, a time-variant reliability analysis method is developed using failure processes decomposition to transform the time-variant reliability problems to the time-invariant problems for dynamic structures under uncertainties (Yu and Wang, 2018). Moreover, a transferred limit state function technique is proposed for efficiently estimating the dynamic failure probability of the structure with the multiple temporal and spatial parameters (Shi et al., 2017).

Another time-dependent set of engineering design problems requires the estimation of flow. Here we study streamflow in dam network design due to the availability of data. Multi-user water reservoirs design has a crucial impact on providing a consistent water supply for local municipal demands, irrigation needs, electricity generation, and fish preservation (Sechi et al., 2019). Hence, multi-user reservoir design requires a careful operation policy to address the distinct requirements. To design such a reservoir, accurate forecasting models are required to know how much water should be stored in the reservoir how much can be used for municipal, agricultural, and environmental conservation, mainly the water uses of fish (Jung and Kim, 2018).

This information is essential for the reservoir to operate efficiently. Hence, we must choose an appropriate forecasting model with acceptable accuracy by comparing different predictive models. By obtaining accurately estimated inflow and targeting the accurate amount of water that needs to be stored in the reservoir, we can decrease the water loss due to evaporation

(Worland et al., 2018). Therefore, each period's estimations can be used to set a better water system design strategy and management policy to regulate the release and storage times on Buffalo Reservoir. However, after a critical evaluation of the literature, we find a gap in predicting the water inflow on the Red River basin and Buffalo Reservoir design. Therefore, in this chapter, our objective is to find a model of the highest accuracy by comparing MARIMA, ANN, and RF (Random Forest); besides, we aim to identify the most impactful time lag and prior hydrological status of the water inflow.

Lag is the delay or time step in the timeline. Consider a discrete sequence of values. For Lag 1, we compare our time series with a lagged time series. In other words, we shift the time series by one before comparing it with itself. For lag 2, we compare our time series with two lagged time series. In other words, we shift the time series by two before comparing it with itself. With the proposed model, we provide a tool for water resource management and identify the significant, influential factors for Buffalo Reservoir water inflow.

Determining the appropriate reservoir operation policy relies on accurate flow measurement, accurate flow simulation, or accurate approximation of water flow into a reservoir (Li et al., 2016a). However, as shown in Figure 4.1, accurate measurement of the water flow into a reservoir using physical experiments is expensive, and accurate water flow simulation is time-consuming and computationally expensive. So, we use surrogate modeling alternative statistical models - models of models or metamodels – for expensive computer simulations and physical experiments to manage the computational complexity.

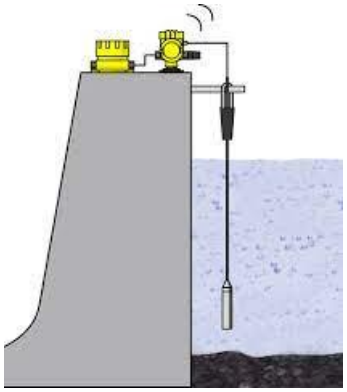
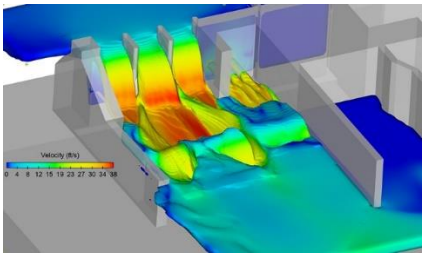
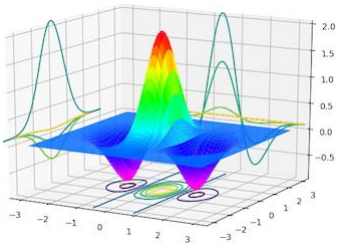
Physical Experiment (Physical Measurement)	Simulation (Hydraulic Simulations)	Surrogate Model (Time Series Analysis)
		
Expensive Time Consuming	**Time Consuming**	**Reduced Cost and Time**

Figure 4.1. Surrogate modeling in dam flow measurement.

As shown in Figure 4.1, accurate measurement of the water flow into a reservoir is expensive and accurate simulation of the water flow is time consuming. So, we use surrogate modeling, which are alternative statistical models - models of models or metamodels – for expensive computer simulations and physical experiments to manage the computational complexity. According to the forecast circumstances and available information, various surrogate models can be used for various forecasting situations. While certain models often provide high predictive accuracy, as of yet, there is no widespread consensus for forecast models across different stream flow scenarios or basins.

As such, it is necessary to analyze the suitability of multiple stream flow forecast models for a particular forecast circumstance. Based on the literature, three classes of the prediction models of time series analysis models, including Multivariate Autoregressive Integrated Moving Average (MARIMA), RF, and Artificial Neural Networks (ANNs), are used in forecasting water planning variables are analyzed. Afterwards, taking the accuracy measures into account, the most precise predictive model is proposed together with an effective method for selecting efficient estimators. In much previous research, implementation of these forecast

methods has been studied in different engineering prediction problems including stream flow predictions. Two popular machine learning prediction models, ANN and RF, have been widely used to approximate various water resource planning goals like the spreading coefficient of natural streams, evaporation loss, earthquake likelihood, conservation water wildlife, and global solar radiation.

For example, Adamowski (Adamowski, 2008) compared the accuracy level of various forecast methods in a daily water demand forecast problem in in the city of Ottawa, Canada. He investigates several ANN, time series analysis, and multiple regression methods to predict maximum day to day water needs. He used data for a ten-year period and variables such as the highest recorded temperature and the highest water need for each day. He found that artificial neural networks perform better than other models in terms of accuracy and the robustness of the results in the highest daily water need compared with the other time series analysis and multiple linear regression models (Adamowski, 2008) . In some of the forecasting problems, a combination of artificial neural networks is used to capture the existing nonlinear relationships between the predictors and predicted values.

Fundamentally, building a precise prediction model using artificial neural networks depends on (i) the right choice of variables to use in forecasting and (ii) validating the created models after the training process. Coulibaly and co-authors (Coulibaly et al., 2000) analyzed how different training methods can affect the accuracy of the artificial neural networks. He found that there are no hard or soft rules to prescribe an exact training/validating process for a specific problem. Nevertheless, in many cases such as (Imrie et al., 2000; Othman and Naseri, 2011), a general ANN with LMBP[3] construction and method are used to achieve a general and comprehensive way of building ANNs. For example, Valipour and co-authors look for an

[3] Levenberg–Marquart Back Propagation

optimal scheme to build ANNs for forecasting a reservoir stream flow on an Iranian reservoir planning problem (Valipour et al., 2012). Some researchers show that concurrently using the efficient stream flow-lagged values together with the other efficient estimators would enhance the approximation accuracy. They use combinations of ANNs with other forecasting models (for example SSA[4], WMRA[5], and MA[6]) to enhance the stream flow forecast accuracy according to the data from different reservoirs in China. Based on their comparison, they found that integration of ANN with moving average has the highest accuracy (Wu et al., 2009).

ANFIS[7] is another forecasting method to create runoff and stream flow prediction models. Most of the studies in which ANFIS has been utilized are conducted on hydrologic prediction models. For example, a reservoir's inflow on the Nile River Basin is estimated in (Atsalakis et al., 2007; El-Shafie et al., 2007). 130 years of historical data was used to build an ANFIS model to predict daily stream inflow. During this study, they show that based on the forecast accuracy measure, ANFIS has a better prediction accuracy in comparison with the earlier ANN-based prediction model. Pramanik and co-authors (Pramanik and Panda, 2009) compare the performance of ANFIS with the performance of the ANN through estimating the outflow from a reservoir in Mahanadi River Basin on a daily basis using the outflow information of the other reservoirs on the upstream. They find that the accuracy of the ANFIS is consistently higher than the accuracy of the ANN which means the ANFIS results are more robust with less variation. Similar results are obtained by other studies. For example, Talebizadeh and co-authors (Talebizadeh and Moridnejad, 2011) reported the same findings in a study on fluctuations of the water levels on the Urmia Lake. However, it is not a universal result, and

[4] Singular Spectrum Analysis

[5] Wavelet Multi-Resolution Analysis

[6] Moving Average

[7] Adaptive Neuro Fuzzy Inference System

some researchers reported the opposite. For example, Karimi and co-authors (Karimi-Googhari and Lee, 2011) achieved higher accuracy from ANN than ANFIS.

ANN has also been compared with other methods. For example, in India, Jain and Srivastava (Jain and Srivastava, 1999) estimate the Indravati Reservoir's stream flow on a monthly basis using autoregressive integrated moving average and ANN. They find ANN more accurate than the ARIMA. Another example is in Iran where Mohammad and co-authors (Mohammadi et al., 2005) estimate the a reservoir's stream inflow using the same methods and find ARIMA more accurate due to their objective which was, daily predictions. Based on their finding, ARIMA leads to better accuracy than neural network in daily inflow predictions.

Climate changeability metrics like uncertain daily, weekly or monthly precipitations are other influential indices which distinguish the performance of the forecast methods. For example, Muluye and Coulibaly (Muluye and Coulibaly, 2007) combine time series data of stream flow along with the climate to increase the forecast accuracy, to estimate the seasonal stream flow to the reservoir from a watershed.

The runoff regime is in the area of study is mainly stream flow dominated system. Thus, we choose and use the most influential physical estimators from the forecasting model development process. Other researchers also use the same physical variables. For example, Shukla and co-authors (Shukla et al., 2013) studied diverse predictors which influence the hydrologic predictability of the land surface.. So, taking the characteristics of the study area into account can lead to proper selection of the hydrologic forecast model.

As can be seen from the literature, we need accurate forecasting models to know how much water should be stored in the reservoir, how much can be used for municipal, agricultural and environmental conservation, which is mainly the water use of fish. Using this information, we can make the operation of the reservoirs effective and efficient. Hence, we need to choose an

appropriate forecast model with acceptable accuracy through comparing different predictive models from the literature to effective operation of the Buffalo Reservoir on the Red River Basin. By obtaining precisely estimated inflow and targeting accurate amount of water which needs to be stored in the reservoir, we can decrease the water loss due to evaporation. Therefore, the estimations for each time period can use to set the better water management strategy to regulate the release and storage times on Buffalo Reservoir.

After critical evaluation of the literature, we find that there is a gap in using ensemble of models in time series analysis and stream flow prediction. So, we propose random forest as ensemble of models. Also, we find that there is no way in time series analysis to find the actual model for calculating the amount of the stream flow. Therefore, we propose MARIMA, ANN and random forest as surrogates of actual stream flow functions.

The rest of the chapter is organized as following, in Section 4.2, we explain the study area and the data which is utilized. In Section 4.3, we explain the forecasting models that we use for the stream flow prediction. Then, in Section 4.4, we discuss the results that we obtain from each predictive model and quantity of stream flow for a specific period of forecasting. Finally, in Section 4.5, we summarized the findings along with the closing statements. The accurate stream flow prediction model helps to more accurate daily operation of the reservoir. Also, when values for the short-term forecast are used in provision of the better operation plans and the observed day to day estimates are used to improve the forecasting accuracy.

4.2 Red River Basin and Available Data

As shown in Figure 4.2, there are thirty-eight reservoirs throughout the Red River Basin. In this chapter, we aim to predict the daily inflow of one of the reservoirs, namely, the “Buffalo Reservoir," in this reservoir network on the Red River Basin. The water in the Red River Basin is used for multiple purposes, including generating electricity delivering water to North Texas,

North Louisiana, Southern Oklahoma, Kansas, Eastern New Mexico, Missouri, South Arkansas, and Colorado. The basin has two reclamation reservoirs, Lugert-Altus and Tom Steed. Reclamation reservoir is the largest wholesaler of water in the country. These type of water reservoirs provide water to more than 31 million people and provide one out of five Western farmers (140,000) with irrigation water for 10 million acres of farmland that produce 60 percent of the nation's vegetables and 25% of its fruit and nut crop. Reclamation is also the second largest producer of hydroelectric power in the western United States. Its 53 power plants annually provide more than 40 billion kilowatt hours generating nearly a billion dollars in power revenues and produce enough electricity to serve 3.5 million homes (USBR, 2022b).

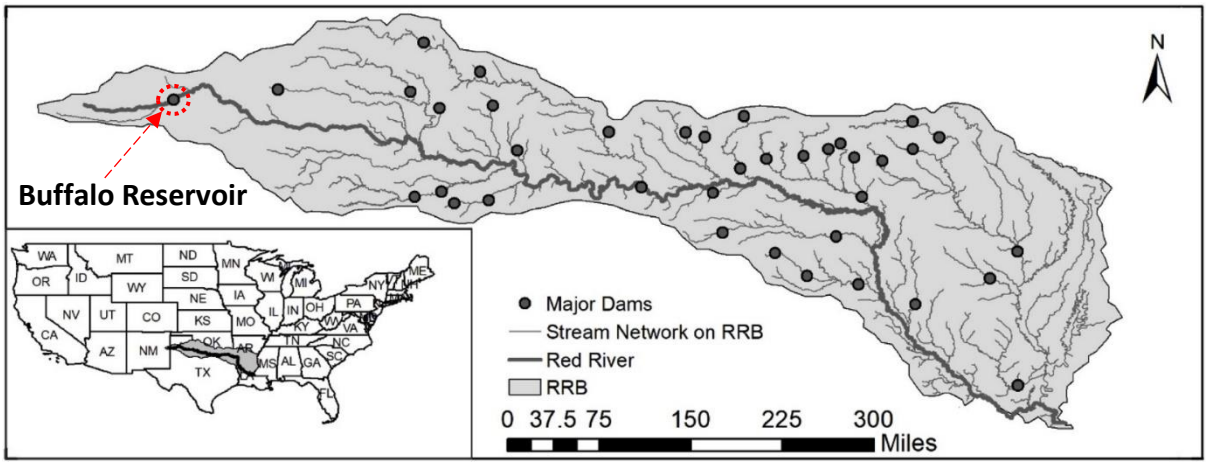


Figure 4.2. Positions of the 38 dams over the red river basin (Zamani Sabzi et al., 2019).

Lugert-Altus and Tom Steed reclamation reservoirs provide 99% of the freshwater sources for about 45,000 people and agricultural water supply for about 48,000 acres of land. The water supply needs in the study area are both immediate and severe. This is because of water quantity, the quality of water, and aging infrastructure. An intense drought has been in effect in this area since 2011, and both Lugert-Altus and Tom Steed Reservoirs are at record lows. In 2015, the red river basin experienced a severe drought followed by flooding; both of these events impact people, planet, and profit. Therefore, managing the supply and sensible distribution of fresh

water to support human activity while sustaining vigorous, effective ecosystems is a major ecological challenge (Poff et al., 2016).

We use data from a recent large-scale, comprehensive analysis of the hydrology, societal water usage, and water availability for the Red River by Xue (Xue et al., 2016) and McPherson (McPherson and Kellogg, 2016) with their co-authors. The current water management protocols may cause substantial water shortages due to evaporation. Therefore, retaining water in reservoirs is necessary as it enhances the probability of meeting the water demand in the Red River Basin. However, a decision support scheme for managing the water resources that stabilize the demand for consistent water resources in the face of evaporation loss has significant advantages. This decision support scheme is used when there is uncertainty in the climate condition and dynamic water circulation.

We need to know the accurate amount of streamflow for each reservoir to plan on the water storage and use for different purposes, including agricultural, industrial, municipal, and environmental conservation, which is the water demand of fish. To find the accurate amount of streamflow, we need to find what data and which predictive model to use. Therefore, we use precipitation, evaporation, minimum and maximum temperature, and the inflow time series data as predictors to predict the inflow rate in Buffalo Reservoir in seven days.

In the next section, we explain the data set. In this chapter, we aim at predicting the water inflow in Buffalo Reservoir on the Red River Basin. We use five different predictors for this purpose: previous inflow of the reservoir, amount of precipitation, amount of evaporation, and maximum temperature and minimum temperature. In addition, we also use Multivariate Auto-Regressive Integrated Moving Average (MARIMA) (Pektas and Cigizoglu, 2013), RF (Herrera et al., 2010), and ANN. Based on our gap analysis in Section 4.1, ANN and RF are machine learning ensembles of the surrogate method, which are effective when we have non-

stationary data with non-linear behavior. MARIMA is the only classic time series analysis method used when we obtain extremely low out-of-sample accuracy.

Our problem has a unique feature with both independent variables (e.g., precipitation, temperature) and dependent variables (inflow in the previous day and other time-related variables). We use the time series model, MARIMA, as a Variable Autoregressive (VAR) analysis method, as discussed in Section 3.1. We analyze our problem as a single, specific time series prediction model using this method. We also use a completely different approach of non-linear machine learning models, ANN and RF, to deal with the data nonlinearity and fit them to our time series. Finally, we divide the data into two sets to validate the results, one training dataset and one test data set.

4.3 Method for Predicting Water Inflow in Buffalo Reservoir

The problem analysis is performed by conducting time series analysis as surrogate models through MARIMA, RF, and ANN. The recorded historical data are used for streamflow forecasting models. Comparing the accuracy of the built models, the model with the highest accuracy is used to estimate the streamflow. As shown in Figure 4.2, our method consists of three main phases, including Data Understanding and Preparation (Phase 1); Modeling (Phase 2); Comparison between Models based on R-Squared and RMSE (Phase 3).

Phase 1: In Phase 1, we find the available data on Buffalo Reservoir (Gaitan et al., 2016). Based on its relevancy and feasibility for time series analysis and streamflow prediction, we use all the five variables in Figure 4.3 as predictors, and the dataset is clean without any missing values or outliers. We use the data modelled by global climate models (GCMs), which are the outputs from the Coupled Model Inter-comparison Project Phase 5 (CMIP5) (Taylor et al., 2012). The CMIP5 consists of more than 50 GCMs from 20 modeling centers which were used to replicate the past and simulate future climate throughout the Red River Basin. The time

period for the past replications of GCM is from 1900 to 2005 and the time period for the forecasting simulations of GCM is from 2006–2099 (Flato et al., 2014).

The first predictor is the stream inflow to the Buffalo Reservoir, a numeric variable ranging from zero to 629 acre-feet per day. Flow is an input variable when it is used for the current period. It is also used as an output variable with different time lags. For example, we use one-time lag, which means we use the flow into Buffalo Reservoir from one previous period. We also use two-time lags, which means we use the flow into Buffalo Reservoir from the only flow of the second previous time lag. We continue this until the seventh time lag, using the inflow from only the seventh period before the current period. The second predictor is the precipitation that goes into of the Buffalo Reservoir. It is a numeric variable ranging from 0 to 95.30 acre-feet per day, and it is an output variable.

The third predictor is the evaporation from the Buffalo Reservoir each day. This variable ranges from 0.007 to 0.39 acre-feet and is used as an out output variable. The fourth and the fifth predictors are the maximum and minimum temperature in the Buffalo Reservoir, which ranges from -22.98 ºF to 27.19 ºF and are output variables.

Table 4.1. Data description

Variable	Type	Description
Flow	Numeric/input and output	Between 0 and 629 (acre-feet/day)
Precipitation	Numeric/input	Between 0 and 95.39 (acre-feet/day)
Evaporation	Numeric/input	Between 0.007 and 0.39 (acre-feet/day)
Maximum Temperature	Numeric/input	Between -22.98 and 27.19 (Fahrenheit)
Minimum Temperature	Numeric/input	

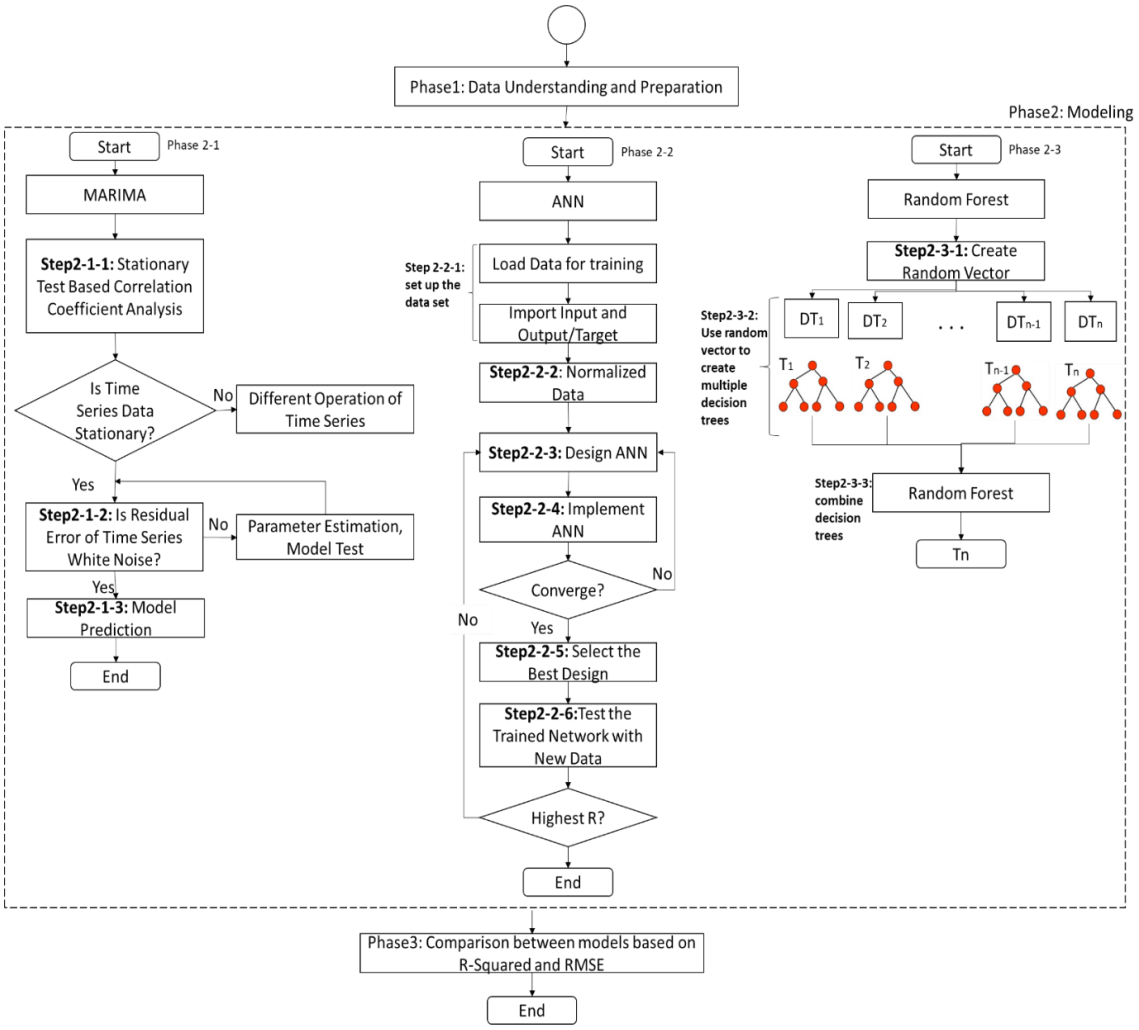


Figure 4.3. Explanation of the method

Phase 2: in Phase 2, we create three types of predictive models, MARIMA, ANN, and RF.

Phase2-1 (***MARIMA modeling***): As shown in Figure 4.3, we first create the MARIMA model. Formulating the MARIMA model is summarized in Figure 4.3, Phase 2-1. The steps for building the MARIMA model are as follows:

Step 2-1-1: We first implement the static test on the data. We use AR and MA operators to identify whether the time series is a stationary sequence. If not, we difference the data to make it stable. The maximum likelihood estimation method is used to evaluate the parameters in MARIMA. Then the resulting model is tested.

Step 2-1-2: in this step, we diagnose whether the residual error of the time series is white noise or not. A time series is a white noise if the variables are independent and identically distributed with a mean of zero. By white noise, we mean that all the predictors have the same variance (σ^2), and every value has a zero correlation with the other variables in the time series. If it is not, we re-estimate the parameters until the generated residual error becomes white noise.

Step 2-1-3: Finally, MARIMA is used for prediction with suitable parameters.

Phase 2-2 (ANN Modeling): As shown in Figure 4.3, we create the ANN model after building the MARIMA model. The procedure to formulate the ANN model is summarized in Figure 4.3, Phase 2-2. The steps for building the ANN model are as follows:

Step 2-2-1: We load the data and identify the input and output variables in this step. Our problem identifies precipitation, evaporation, maximum and minimum temperature, and the previous period (lag) flow as input variables. Also, we define the flow for the current period as the output variable.

Step 2-2-2: In this step, we normalized the data so that all the inputs are at a close range.

Step 2-2-3: In this step, we design the ANN, with (1) one input layer with five neurons as input variables; (2) one and two hidden layers, here we chose one hidden layer design after comparing the results; (3) one output layer with one output.

Step 2-2-4: We implement the ANN algorithm with the imported data and create architecture in Step 3. We then use the training data to train the model.

Step 2-2-5: In this step, if the created ANN model has converged, we calculate the MSE and RMSE for the training data. We selected the design with one input layer with five input variables, one hidden layer, and one output layer with one output variable.

Step 2-2-6: We test the trained network with the test data in this step. Then we calculated RMSE and R-Squared and use them to compare with MARIMA and RF models results.

Phase 2-3 (RF Modeling): As shown in Figure 4.3, we create the RF model after building MARIMA and ANN models.

The procedure to formulate the RF model is summarized in Figure 4.3, Phase 2-3. The steps for building the RF model are:

Step 2-3-1: In this step, we load the data and create random vectors using a probability distribution. These probability distributions are calculated based on classes that are difficult to classify.

Step 2-3-2: In this step, a random vector can be incorporated into the tree-growing process in multiple ways. The leaf nodes of each tree are labeled by estimates of the posterior distribution over the data class labels. Each internal node contains a test that best splits the data space to be classified. A new, unseen instance is classified by sending it down every tree and aggregating the reached leaf distributions. Each node is split best among the randomly chosen subset of predictors at that node instead of the complete set.

Step 2-3-3: In this step, we ensemble the decision trees created in Step2.

Phase 3: We build a MARIMA model using only a one-time lag and forecast only one period with MARIMA since we cannot forecast more than one step. We build one type of ANN, but we forecast the seven subsequent periods with that. Also, to examine the importance of different time lags, we build seven types of the random forest using the set of lags of (1, 2, 3, 4, 5, 6, 7). We find that the time lag of 7, which means using the seventh day before the day that we forecast for it, estimate the next days' flow, captures the nature of the problem more accurately than other time lags. So, we use the random forest with seven lags, but we forecast the next seven days' flow with that. Then, in Phase 3, we compare the results of MARIMA, ANN, and random forest, based on two accuracy measures, R-Squared and Root Mean Square Error (RMSE). We use the R software and packages "for," "MARIMA," and "RandomForest" to implement the models.

4.4 Predicting the Inflow in Buffalo Reservoir using Multivariate AutoRegressive Integrated Moving Average (MARIMA)

MARIMA is a multivariate method to analyze time series introduced by Box and co-authors (Box et al., 2015). Usually, MARIMA models are generalized types of the univariate ARMIA models (Box et al., 2015). However, ARIMA has the issue of overfitting (Bennett et al., 2014), so, we use MARIMA to avoid this issue. The non-seasonal ARIMA is MARIMA (d, p, q), where d is the number of lags of the moving average, p is the number of orders of differencing, and q is the number of moving average lags. In this chapter, the three phases of MARIMA (e.g., identification, approximation, and diagnosis) are used to obtain the model with the highest accuracy to estimate streamflow. Suitable values for p and q are found by analyzing the

autocorrelation ACF[8] and PACF[9]. Shukla and co-authors (Shalamu, 2009) and Abudu and co-authors (Abudu et al., 2010), and Zamani and co-authors (Zamani Sabzi et al., 2018) used a univariate ARIMA in the seasonal form to forecast the streamflow for each month.

A MARIMA model is defined by Equation 4.1 to Equation 4.5 as follows:

$$\gamma(A)y_t = \alpha(A)b_t \qquad \text{Equation 4.1}$$

Where y_t is the dependent variable of the amount of inflow in time *t*, A is the lag, b_t shows noise.

$$y_t = 1 - A^d Y_t \qquad \text{Equation 4.2}$$

as stationary series,

$$\gamma(A) = 1 - \gamma_1 A - \gamma_2 A^2 - \cdots - \gamma_p A^p \qquad \text{Equation 4.3}$$

as non-seasonal autoregressive series,

$$\alpha(A) = 1 - \alpha_1 - \alpha_2 A^2 - \cdots - \alpha_q A^q \qquad \text{Equation 4.4}$$

As for non-seasonal moving mean series, q stands for the order of the moving mean series, such that p shows the auto-regression series order. Then, the lag A is computed by Equation 4.5:

[8] Autocorrelation Function

[9] Partial Autocorrelation Functions

$$AY_t = Y_{t-1}$$ Equation 4.5

The data we have are not stationary due to the unstable variance and mean of non-seasonal day-to-day data of streamflow. Therefore, to forecast the daily streamflow, we need to stabilize the data to use in the MARIMA model. In other words, we need to implement adequate differencing to stabilize the time series' average and variance.

MARIMA model has three parameters: *p*, which identifies how many previous lags or time steps – in these problems days – must be input; *q*, which determines the subsequent observation is the average of how many previous lags or time steps (here days); *d*, determines how many nonseasonal differences are required to achieve stationary data.

The parameter *q* is calculated by plotting the PACF, and the parameter *p* is calculated by plotting the ACF. D is found by trying different differencing terms and choosing the value for *d*, which results in the lowest variance. The autocorrelation is less than -0.5 which is a very low variance (Brockwell and Davis, 2009). The parameters that produce the highest accuracy of the MARIMA model are *p*=7, *q*=1, and *d*=1 and we can use the most appropriate lag set (1, 2, 3, 4, 5, 6, 7. This is the most vital auto-regression parameter.

4.5 Predicting the Inflow in Buffalo Reservoir Using RF

We use RF because trees are weak learners and unstable. However, the observed predictive accuracy is improved through an ensemble of many decision trees, as in RFs. In other words, they may cause a minimal error with considerable variance. One of the approaches to boost the

functionality of DT[10] is to integrate the responses of various decision trees. In an RF model, we train a lot of decision trees such that every tree is trained through a random feature subset (Breiman, 2001). This helps de-correlate the individual trees of the forest. We define an RF, an ensemble of decision trees, as $\boldsymbol{R} = \{\boldsymbol{T_k}\}_{\boldsymbol{k}\in[\boldsymbol{m}]}$. The RF is used to calculate the average of single decision tree outcomes, thus:

$$(\boldsymbol{y}) = \frac{\mathbf{1}}{\boldsymbol{m}}\sum_{\boldsymbol{k}\in[\boldsymbol{m}]} \boldsymbol{T_k}(\boldsymbol{y}) \qquad \text{Equation 4.6}$$

A random sample of *m* features is set at each node to create the RF. Only these m features are considered for splitting. Typically: m = $\sqrt{p}$ or $m = \log_2 p$ (p is the number of features). Then, we try to improve bagging, which is accomplished with an ensemble algorithm that fits several models of various samples of a training dataset, then combines all models' predictions - "de-correlating" the trees. Forming an RF is an expansion of bagging which also choses subsets of features used in all samples randomly.

Consider one strong predictor in our data set which reduces a measure of error the most. All our trees tend to make the same cuts because they all share the same features. This makes all these trees look remarkably similar, hence increasing correlation. We allow the random forest to choose only several predictors in performing the split randomly to solve tree correlation. Now the trees all have different randomly selected features to perform cuts on. Therefore, the feature space is split into different predictors, de-correlating all trees to decrease variance.

[10] Decision Tree

Third, the RF is built by assembling these decision trees. Using RF instead of a single decision tree decreases the interpretability but usually increases the accuracy of the ultimate result. Finally, we want to see if an ensemble of trees in the configuration of RF performs better than classic time series analysis methods like MARIMA.

4.6 Predicting the Inflow in Buffalo Reservoir using Artificial Neural Network (ANN)

We use ANNs as a machine learning technique. ANNs are predictive techniques created based on the neural structure of the human brain. These models can be used to recognize the variables' complicated and non-linear relationships. Usually, ANNs are modeled via multiple layers of interconnected neurons and calculated using historical data and future values. In this chapter, we use a common type of ANNs, the FFNNs[11].

In Figure 4, a classic construction of the ANN computation function is shown with a layer for input, another layer for output, and a hidden layer. In such a 3-layer ANN, we calculate the output variables by turning the non-linearity of the problem into the linear arrangements of the input elements. Finally, the outputs are clearly articulated and computed via Equation 4.7:

$$\hat{y}_j = f_0[\sum_{i=1}^{N} z_{ji} * f_g\left(\sum_{j=1}^{M} z_{ij}x_j + z_{i0}\right) + z_{j0}] \qquad \text{Equation 4.7}$$

when the variable of z_{ij} stands for the most appropriate allocated weight, which links the j_{th} element in the network in the first layer (input layer) to the i_{th} element on the second layer

[11] Feed-forward neural networks

(hidden), z_{i0} is the measured error, which is allocated to the i_{th} element on the second layer, f_g represents the so-called "activation function" in the second layer, z_{ji} is the measured weight which connects the i_{th} element on the second layer to the j_{th} element on the layer output, z_{j0} is the error amount allocated to the j^{th} element on the layer of output, and f_0 is the output activation function such as the following:

$$S(m) = 0.5 \sum_{q=1}^{M} \sum_{j=1}^{P} \left[x_{qj}(m) - \hat{x}_{qj}(m)\right]^2 \qquad \text{Equation 4.8}$$

in which M is the number of observations (inputs) used, P is the number of estimated output variables, $x_{qj}(m)$ is the observed variables (target), and $\hat{x}_{qj}(m)$ is the projected value for the j^{th} element by m^{th} replication. One of the functions which are built for assessing the accuracy of the created ANNs is:

$$I_{t+1,t+2,t+3,t+4,t+5,t+6,t+7} = f(I_t, E_t, P_t, T_{Max}, T_{Min}, M_j) \qquad \text{Equation 4.9}$$

such that $I_{t+1,t+2,t+3,t+4,t+5,t+6,t+7}$ is the estimated streamflow on day to day basis at 1 to 7 days before; I_t is the observed day to day amount of flow at the given day of I_t; E_t is the evaporation from the Buffalo Reservoir; P_t is the precipitation into the Reservoir and T_{Max} is

the maximum recorded temperature, and T_{Min} is the minimum recorded temperature, M_j is the

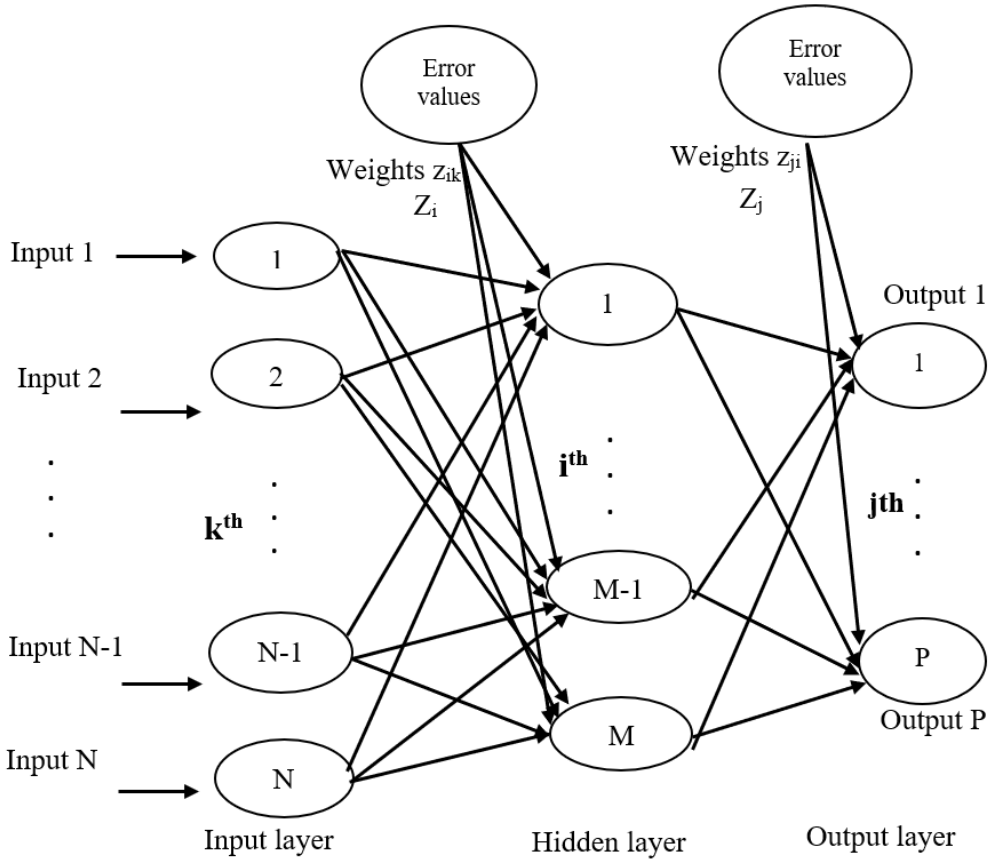


Figure 4.4. Typical construction of an ANN with a single layer of input, hidden, and output.

4.7 Results and Discussions of Predicting the Inflow in Buffalo Reservoir

We divide the data into training and test sets to compare the predictive models based on their accuracy performance—30% of the dataset between 2002 and 2018 is used to as a test dataset. As shown in Table 4.2 and Table 4.3, we build one type of MARIMA using only a one-time lag. We forecast only one period with MARIMA since we cannot forecast more than one step with MARIMA. We build one type of ANN, but we forecast seven subsequent periods with that.

Also, to examine the importance of different time lags, we build seven types of RF using the set of lags of (1, 2, 3, 4, 5, 6, 7). Comparing the results of MARIMA, ANN, and RF, we see that RF has the highest R-Squared values for all the prediction scenarios, as shown in Table 4.2. Also, as shown in Table 4.3, the RMSE of the RF model is lower than the ANN and MARIMA models.

MARIMA, ANN, and RF, as shown in Table 4.2, are compared based on the accuracy performance in forecasting. We use the $E_t, P_t, T_{Max}, T_{Min}$, and I_t to create the models and estimate the next step. In Table 4.2, we provide the forecasted inflow amount for I_{t+1}, I_{t+2}, I_{t+3}, I_{t+4}, I_{t+5}, I_{t+6}, I_{t+7}. RF has the lowest RMSE in forecasting the streamflow to Buffalo Reservoir, as illustrated in Table 4.3. MARIMA is the next most accurate method for predicting one-day stream, but it is only used to predict one day lag. In Table 4.2 the forecasting accuracy performance of the different utilized methods is shown. To compare accuracy, both RMSE and correlation measures are used.

Table 4.2. Performance evaluation and comparison between MARIMA, ANN and RF models.

	R^2 Values						
Model Type	**lags**						
	I_{t+1}	I_{t+2}	I_{t+3}	I_{t+4}	I_{t+5}	I_{t+6}	I_{t+7}
MARIMA	0.923	-	-	-	-	-	-
ANN	0.895	0.799	0.522	0.398	0.358	0.356	0.356
RF	0.947	0.948	0.947	0.948	0.949	0.948	0.982

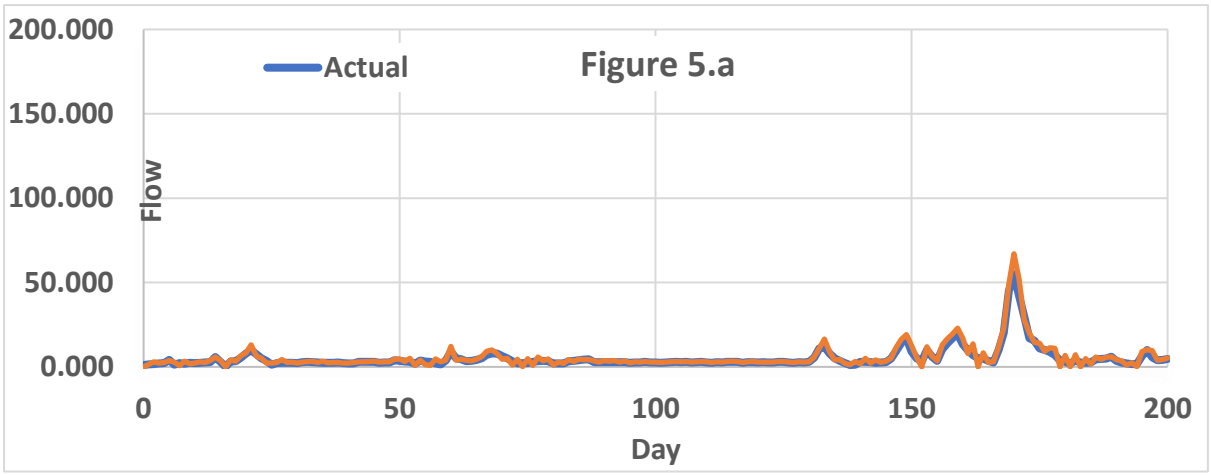
Although the R^2 of the daily forecasting model (which is in the first row in Table 4.2 and used to forecast the next day) is marginally lower than the chosen RF model (3rd row of the

numerical findings in the RF model of Table 4.2), it is not chosen as the most appropriate model because the forecast for the next week (I_{t+7}) has a higher R-Squared. Also, the daily MARIMA forecasting model is acceptably accurate. However, MARIMA is a suitable method for a weekly forecast. We illustrate the forecasted amounts versus the observed amounts in Figure 4.5 to Figure 4.7.

Table 4.3. Accuracy of the used forecast method in various forecast time horizons

Accuracy measure	I_{t+1}	I_{t+2}	I_{t+3}	I_{t+4}	I_{t+5}	I_{t+6}	I_{t+7}
RMSE (MARIMA)	9.444	-	-	-	-	-	-
RMSE (ANN)	8.324	8.235	8.264	8.298	8.165	8.268	8.458
RMSE (RF)	7.15	7.28	7.22	7.26	7.25	7.24	7.24

While the correlation coefficient of the daily forecasting model (which is in the first row in Table 4.2 and used to forecast the next day) is marginally lower than the chosen RF model (3rd row of the numerical findings in the RF model of the Table 4.2), it is not chosen as the most appropriate model because the forecast for the next week (I_{t+7}) has a higher R-Squared.



.

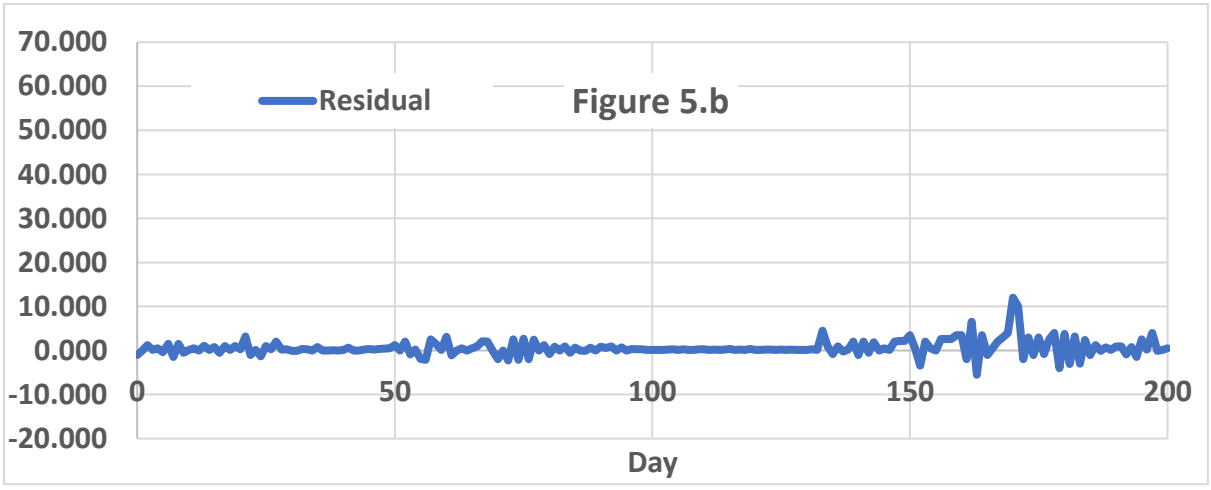
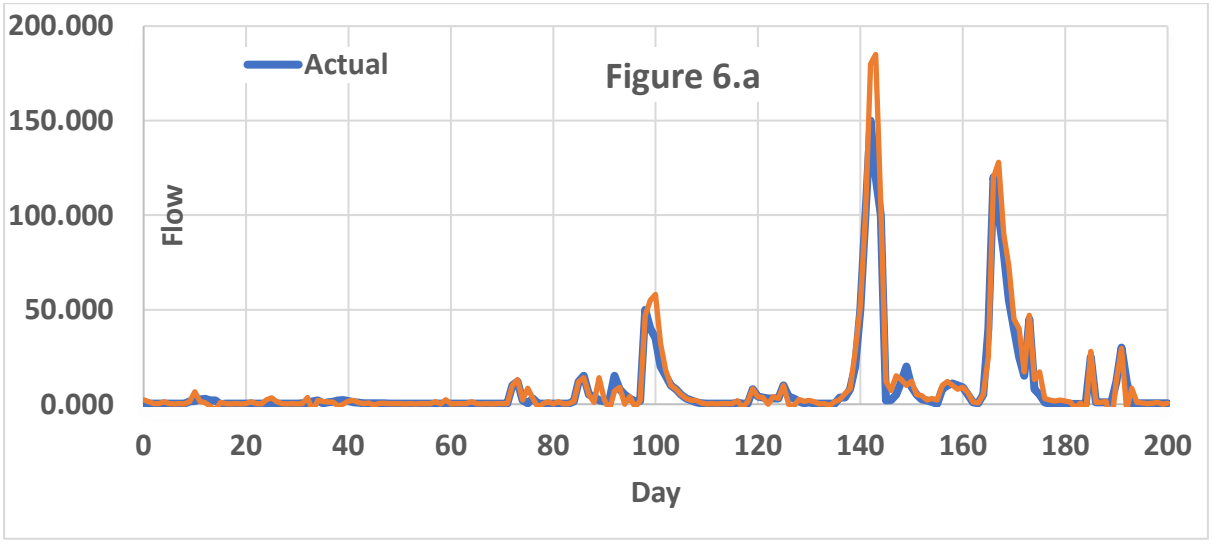


Figure 4.5. Visualization of the predicted, actual, and residual values.
a) Predicted values vs. observed values using MARIMA. b) Residual plot over the time. 30% of the data between 2002 and 2018 is used for testing the results.

Since the chosen models of RF, MARIMA and ANN are used to forecast for the subsequent seven days, and only with MARIMA model we predict the next one day on daily inflow forecasting models. Also, daily MARIMA forecasting model is meaningfully accurate, however, because of its larger error, MARIMA is not selected as a suitable method for weekly forecast. We illustrate the forecasted amounts versus the observed amounts in Figure 4.5 to Figure 4.7.



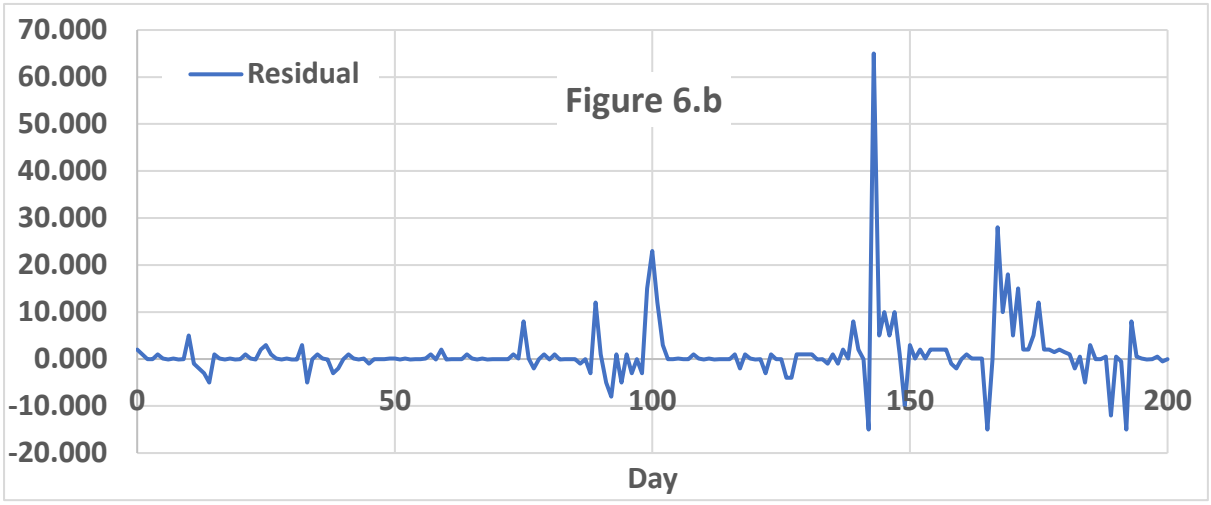
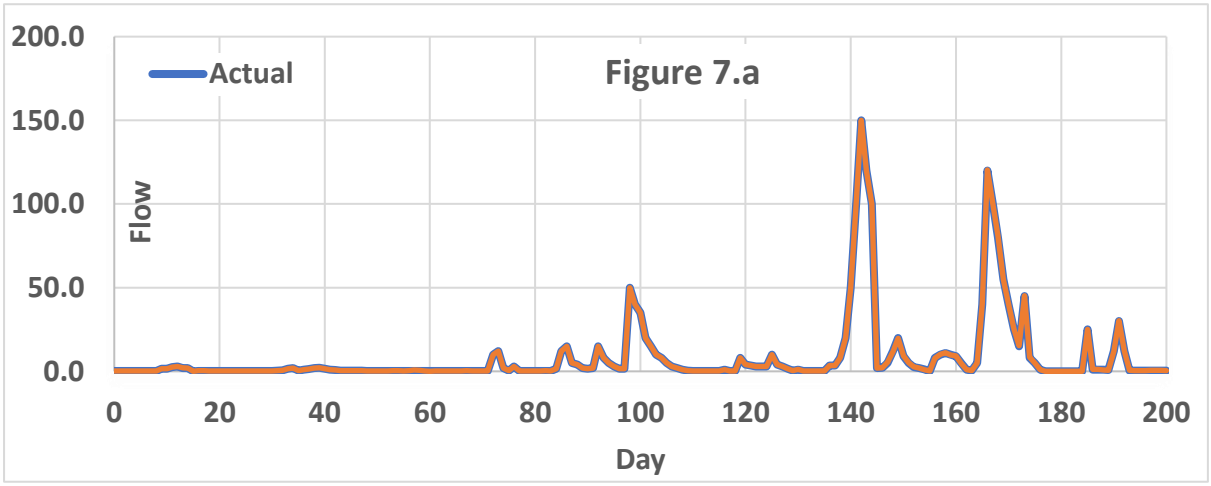


Figure 4.6. Visualization of the predicted, actual, and residual values.
a) predicted values vs. observed values using ann. b) residual plot over the time. 30% of the data between 2002 and 2018 is used for testing the results.



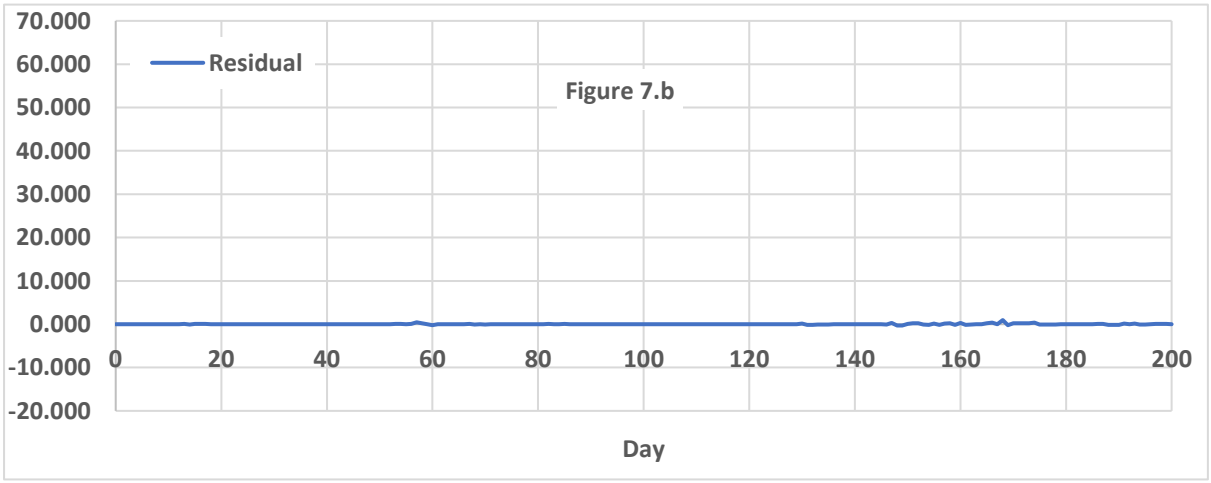


Figure 4.7. Visualization of the predicted, actual, and residual values.

a) Predicted values vs. observed values using RF. b) Residual plot over the time. 30% of the data between 2002 and 2018 is used for testing the results.

4.8 On Verification and Validation – Theoretical Structural Validity (TSV)

The relationship of these research efforts reviewed in this chapter with the constructs of the systematic approach developed in this dissertation is highlighted in Figure 4.8. In Section 4.1, a critical evaluation of the literature on temporal variables, the ways to deal with them and how to create surrogate models which are able to deal with temporal variables are provided. While in Section 4.2, the data used in surrogate modeling, and water inflow prediction is described.

Creation of MARIMA, RF, and ANN models are explained in Sections 4.4, 4.5 and 4.6 respectively. Results of predicting the inflow in Buffalo reservoir using surrogate modeling and the verification and validation of created models are illustrated in Sections 4.7 and 4.8 respectively.

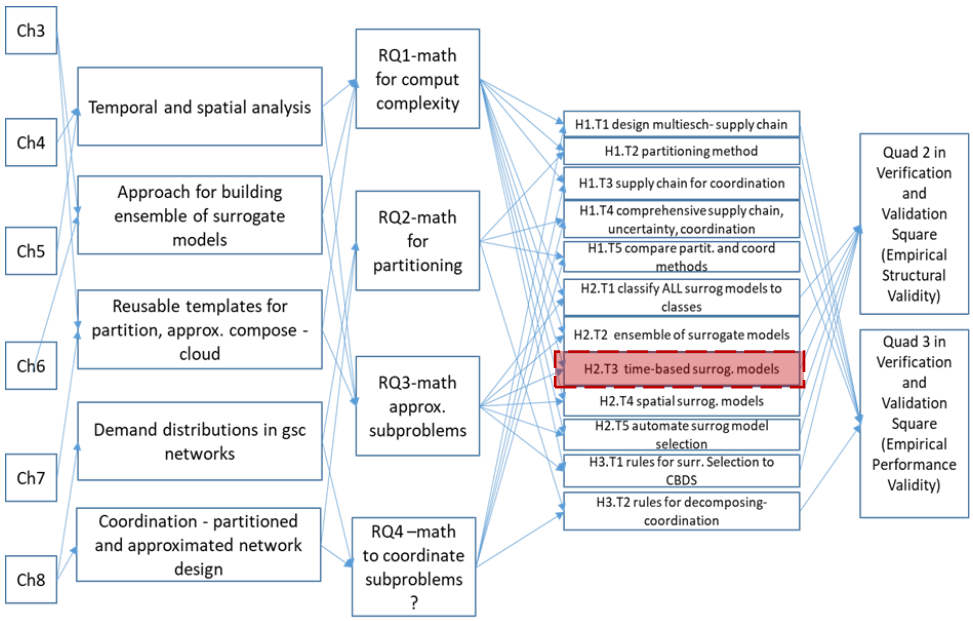


Figure 4.8. Relationship of research efforts with the temporal surrogate models and connection between chapters of the dissertation.

4.8.1. TSV of function-based design (Gap 4 in Figure 4.7)

Theoretical structural validation refers to accepting the validity of individual constructs used in the systematic function-based approach and accepting the internal consistency of the way the constructs are put together. Theoretical structural validation involves systematically identifying the scope of the proposed approach's application, reviewing relevant literature and identifying the research gaps that is existing, identifying the strengths and limitations of the constructs uses based on literature review, determining the constructs and approaches that can be leveraged for the systematic function-based approach while reviewing literature on the advantages, disadvantages and accepted domains of application, and checking the internal consistency of the constructs both individually and when integrated.

In Chapter 4, we establish the generic nature of the systematic approach and why the approach is appropriate for concept generation during early stages of design. By carrying out literature search, it is shown that the systematic surrogate-based approach and the associated constructs have been previously applied for problems in various domains in a successful manner and are verified and validated. The use of these generic systematic approach for dealing with temporal data in the surrogate modeling process is not addressed in past literature. Based on the critical review of literature in Chapter 4, it is inferred that the application of dynamic surrogate modeling is mostly on areas related to mechanical, control, software and process engineering and is mostly applied for selection of materials for replacing computationally expensive simulations.

The theoretical structural validity of the time-based surrogate modeling approach for predicting in temporal datasets to achieve accurate predictions and ensemble od time-based surrogate models which are accepted by the logical procedure of literature review, gap analysis and development and evaluation individual and ensemble surrogates. Empirical studies need

to be carried out to establish the usefulness and effectiveness of the approach and is addressed in Chapter 4.

4.8.2. TSV of temporal surrogate models (Gap 4 in Figure 4.7)

Theoretical structural validation refers to accepting the validity of different features, including temporal used in the surrogate modeling and accepting the internal consistency of the way the different surrogate models are put together. Theoretical structural validation involves systematically identifying the scope of the proposed framework's and design method's application, reviewing relevant literature and identifying the research gaps that is existing, identifying the strengths and limitations of the constructs used based on literature review, determining the constructs and approaches that can be leveraged for the time series analysis and time-based surrogate models and the concept of the ensemble of surrogates while reviewing literature on the advantages, disadvantages and accepted domains of application, and checking the internal consistency of the constructs both individually and when integrated. In Chapters 1 and 2, the need for a surrogate modeling for managing computational complexity is established. The individual surrogate modeling is critically reviewed in Chapter 3 and the functionalities and limitations associated with the methods are established. The limitations of individual surrogate models in terms of the following are discussed: i) when using individual surrogate models, the fitness of the models to the data in the entire solution space is not taken into account, ii) surrogate models have limitations in terms of dealing with temporal and spatial data, iii) there is no systematic and organized way to automatically choose the appropriate surrogate model.

In Chapter 4, Gap 4, which is on Integrating SM and time series, is posed after a critical evaluation of the literature. To address this gap, Research Question 4 is posed as "what is the appropriate SM to use when the data includes time-dependent variables as predictors? Then, to answer

this question, we hypothesized that through replacing the design of experiments with the time lags analysis, we find the SM which is useable in temporal variables. By testing this hypothesis through creation of spatial surrogate models and testing it on dam water inflow prediction problem on Buffalo reservoir on the Red river, we enabled dealing with temporal data by incorporating time series (lag analysis) with SM as the key outcome. Also, we enabled temporal data can be used in SM using different time lags. These SMs are better than classic time series analysis methods like ARIMA, MARIMA and ARIMAX. Also, EOS are better than individual SM in temporal variables.

Now, that we have discussed how to deal with temporal data in surrogate modeling, we are going to talk about how to build mathematics to deal with spatial variables and features in the dataset. This helps us to integrate spatial statistics and specially the idea of geographically weighted regression with surrogate modeling.

4.9 Closing Remarks on Predicting the Inflow in Buffalo Reservoir

Surrogate modeling replaces computationally expensive simulations and costly, complicated physical experiments in many engineering design problems. However, there is a gap in using surrogate modeling in other fields. Dam network design is one field where measuring the flow into the reservoir can be done by physical measurement, which is an expensive, or hydrological simulation, which is very time-consuming and computationally expensive. Therefore, surrogate models as simulations can be used to manage this computational complexity. However, the challenge of using surrogate modeling for flow prediction is that the flow values are time dependent. Also, ensembles of surrogates have been widely used in engineering design. For example, we use the random forest as an ensemble of decision trees in this problem. In this chapter, there are three contributions (1) using surrogate

modeling to predict flow for dam network design, (2) incorporating time series analysis in surrogate models for water network design, and particularly the concept of lag with surrogate modeling in this chapter using different time lags. This is new in surrogate modeling literature; (3) using and ensemble of surrogates in an RF model to improve accuracy and achieve more accurate predictions.

The problem of predicting flow for dam network design has the unique feature of independent variables (e.g., precipitation, temperature) and dependent variables (inflow in the previous day and other time-related variables). We use time series models in this problem using ARIMA to include lags of the dependent variable, stream inflow. However, using this classic time series method for analysis, we obtain an unacceptably low out-of-sample accuracy. The RF method is the most accurate predictive model for this problem. Therefore, selecting suitable surrogate models to create an ensemble of surrogates is another takeaway for this chapter. Also, based on our findings by comparing three types of inflow prediction models, we explain the importance of choosing the appropriate features as predictors of forecast accuracy. For example, discovering and using climate changeability metrics (such as previous inflow values as time-related metrics ($I_{t+1}, I_{t+2}, I_{t+3}, I_{t+4}, I_{t+5}, I_{t+6}, I_{t+7}$) using RFs leads to greater forecasting accuracy for stream inflow forecasting. Data-driven decisions on predictor selection and surrogate model selection are other salient features of our method. Using time-related metrics for daily stream inflow forecasting does not improve the forecast accuracy as well as using the seventh previous time step (lag) forecast. The influence of an existing observed trend is more evident for longer forecast horizons. Consequently, we can create a forecast function using the influential physical estimator. By comparing the accuracy of different models, we find that using more information via an ensemble of models can enhance the forecasting performance.

Our method is useful for other engineering problems, particularly water planning problems, empowering water resource planners to conserve the water to meet agricultural, environmental, and social needs and hydropower energy generation as in all water planning problems, the amount of water is uncertain and needs to be estimated. Overall, preprocessing the monthly historical data to classify the recorded data and accumulating the time-related stream flow pattern metrics enhances the forecast accuracy of the built predictions. We learn that forecast accuracy in both RF and ANNs depends heavily on the choice of the estimators and analysis of the chosen estimators. Greater forecast accuracy is obtained by analyzing the historical data and choosing appropriate time lags. Especially on the prediction of a weekly basis, using RF together with the time-related metrics meaningfully enhance the forecast accuracy.

The methods proposed in this chapter can be used by people who are working on surrogate models, forecast model designers, those who want to embed machine learning in forecasting methods and time series models with surrogate modeling; and practitioners who want to implement expert systems in practical problems. In this chapter, the method has been demonstrated for an environmental management problem, water resource planning. This problem has been used because of the availability of data, however it could also be applied to designing anything in which time dependent flow is an issue, for example, biomedical problems, the management of product sales including the acceptance of new products, and monitoring manufacturing processes as well as products in which fluid flow is an issue.

CHAPTER 5 APPROXIMATION OF REALITY USING SPATIAL SURROGATE MODELING

The objective in this chapter is to introduce how spatial statistics and geographically weighted regression predictive models can be used as surrogate models. We incorporate the geographical neighbourhoods and particularly the concept of correlation of census blocks as a feature with the surrogate modeling in this chapter. We proposed the idea of spatial surrogate models by adding this feature (space/location) as a new feature in the process of surrogate modeling. The findings in this chapter can be used by people who are working on census data and computationally expensive simulations.

Purpose: While the geography of crime has been a focal concern in criminology from the very start of the discipline, the development and use of statistical methods specifically designed for spatially referenced data has evolved more recently. Besides geographical location, demographic information has significant impact in criminology. In this chapter, we aim to discover hidden patterns in crime data and predict the crime rate based on socio-spatial information.

Design method/approach: We use two different geospatial statistics methods; point pattern analysis to discover the hidden spatial patterns within the data, and geographically weighted regression to predict the crime rate. We use Los Angeles city as an example to exercise the designed models. The dataset which is used includes detailed crime reporting data with socio-spatial data.

Findings: We find spatial autocorrelation between demographic features specially gender, age and population and use it to cluster the counties in Los Angeles city. We predict the crime rate in Los Angeles with an acceptable accuracy of $R^2 = 0.8$. Also, we detect meaningful crime patterns and clear clusters with high crime rate in the study area.

Originality/value: Addressing the gap of using social and demographic data in discovering patterns in crime rate. Also, clustering the Los Angeles counties into several communities with mutual criminological features which enables us to prevent the spatial diffusion of the crimes by focusing on high crime rate areas. The proposed model can play a critical role in the efficient allocation of scarce law enforcement resources

5.1 Frame of Reference for Spatial Surrogate Models

The spatial-demographic crime distribution is getting increasing attention from both researchers (sociologists, criminologists, geographers, economists, etc.) and law enforcement agencies (Bernasco and Elffers, 2010; Li et al., 2014; Ratcliffe, 2010; Roth et al., 2013; Tita and Radil, 2010). One of the most under-researched area of criminology is that of socio-spatial crime patterns (Ratcliffe, 2010). It has been clearly highlighted that there is a need to combine geographic and demographic analyses and representation of the data (Roth et al., 2013). Therefore, this is the gap in the literature which we address the present chapter.

In Chapter 5, we aim at addressing such needs by proposing a pattern recognition approach along with geographically weighted regression (GWR) using socio-spatial urban crime count data. We then apply to a dataset consisting of 12 month (2010) counts of crimes for the 1800 census tracts in Los Angles, California. Socio-demographic census tract characteristics and unobserved heterogeneity (random effects) across census tracts.

Another motivation to conduct this research comes from the research of (Gorr and Harries, 2003) and (Cohen et al., 2007) ,that study a range of prediction models for city crimes. When there is no potential socio-economic predictors and spatially lagged response variables, they

find that different kinds of exponential levelling led us to find the best mean prediction point accuracy based on average squared and average absolute ratio prediction error. Relative to these contributions, we benefit from two key advantages. First, as mentioned above, we are able to generalize the forecasting and clustering procedure developed by (Liesenfeld et al., 2016) for latent spatial count models to account also for temporal lags as well as unobserved and observed (socio-economic) heterogeneity. Second, we benefit from access to highly disaggregated and, foremost, internally consistent data at the census tract level combining Uniform Crime Reporting (UCR) data classified according to the handbook of the US Department of Justice (2004) with socio-economic data from the 2000 census.

As a preview of our main results, we test hypothesis of having pattern and crime clusters based on socio-economic descriptions for crime intensities and identified significant endorsement of the 'non-random crime distribution' phenomenon, whereby the concentration of crime in a census tract provides a principal measure for more potential future crimes (Anselin, 2013; Cohen and Gorr, 2005; Cohen et al., 2007; Wilson and Kelling, 2003). More importantly, we can calculate the density and severity of the crimes for the city of Los Angeles to reduce the severe crimes in a specific census tract. These findings emphasize the crucial significance of completely relying on urban spatial dependence but can also provide a valuable means for effective distribution of law enforcement resources.

We also build geographically weighted regressions which demonstrate the remarkable predictive performance of the created model relative to ordinary least squares (OLS) model (a broadly used benchmark). Furthermore, we can implement point pattern analysis (cluster

analysis) tests, not just point projections, but also obtainable forecasting intervals. Finally, we use this point pattern analysis test to produce statistical validation of our clustering schemes.

The chapter is organized as follows. In Section 5.2 we provide a review of the literature on the socio-economic determinants of variations in crime rates across geographic regions and their spatial and temporal dependence. Also, we provide a critical evaluation of the literature on crime studies in Los Angeles city. In Section 5.3 we describe the data. Section 5.4 presents the point pattern analysis method and geographically weighted regression model and in Section 5.5 we illustrate the results of the analysis and discuss the findings and closing remarks are articulated in Section 5.6.

5.2 Predictors and Dependence in Space of Crime Rates

5.2.1. Predictors used in crime rate prediction in Los Angeles city

Practical studies in crime research usually utilize predictive models to describe observed patterns in crime rates within geographic areas with static borders, for example, census tracts (Helbich and Jokar Arsanjani, 2015), counties (Baller et al., 2001), police precincts (Helbich and Griffith, 2016), or census block groups (Wo, 2019). The foundational context includes sociologic concepts of crime consisting of social ecology and location-based models (Pineda-Ríos et al., 2019). Social ecology concepts like social disorder (Persad, 2019) describe the geographical changes in crime severity based on changing social circumstances of the society. Under location-based models, such as repetitive activities (Cornish and Clarke, 2014) and the logical option method (Tuck and Riley, 2017), the geographical difference of crime severities is identified by the connection in demographic and spatial features. These concepts focus on

some measures of structural circumstances which can enable us to project the crime's geographical dispersal. Structural covariates are used in experiential studies are the indicators of population composition (density, distribution and size) (Canter and Shalev, 2017), structure and population of the residents (proportion of different races, age, gender) (Gray and Parker, 2019; Petersen and Ward, 2015; Peterson et al., 2017; Probst et al., 2019), family organization (proportion of female-headed families, separation rate) (Gottlieb and Sugie, 2019), socio-economic features (employment, income) (Aaltonen et al., 2016; Conley and Topa, 2002; Narayan and Smyth*, 2004) and type of residence (houseowner, rental, and vacancy percentage) (Tolnay et al., 1996).

5.2.2. Spatial dependence of crime data

If criminal actions were only identified by the foundational elements which are included in a prediction model, then there must be no spatial relationship beyond that produced by foundational resemblances of areas which are geographically close to each other (Andresen, 2019). However, spatial clustering generally cannot be fully described by conventional indicators of foundational resemblance among geographical areas (Tita and Radil, 2010; Wright and Skubak Tillyer, 2017).

Spatial autocorrelation has gained increasing attention in practical crime studies for two main reasons (Papachristos and Bastomski, 2018): (1) it is statistical and because of that the approximations of the impacts of predictors and their errors can be unstable if the spatial autocorrelation of data is overlooked (LeSage and Pace, 2009). Therefore, studying spatial autocorrelation is crucial when we are evaluating the peripheral impacts of foundational

predictors on crime data. (2) spatial correlation by itself is of high significance in criminology as positive spatial autocorrelation is taken as indication of spatial dispersal of specific kinds of crime (Tolnay et al., 1996). These impacts can show that gangs and criminals are connected together through networks (Faust and Tita, 2019; Nakamura et al., 2019) and cultural behaviors, where violence disperse all over people and spatial units via direct social contacts (Papachristos and Bastomski, 2018; Tita and Greenbaum, 2009).

5.2.3. Critical evaluation of the literature on crime research in Los Angeles

Even though spatial statistics has not been used for crime research in Los Angeles city, expensive criminological research has been conducted over this city. One of the early studies on criminology over LA county is conducted by (Woolsey, 1979) explaining the . The studies on crime in Los Angeles city can be categorized into three groups: 1) criminal/victim features-based crime research: this type of crime research is based on the socio-demographic features on either criminals or victims of the crimes ; 2) crime type-based research: this type of crime research is focused on the type of the crime; 3) methodological crime research: this type of the crime research is focused on creating or using methods and tools from different disciplines in crime research.

With respect to the criminal/victim features-based crime research in Los Angeles city, Rochelle and co-authors introduced a large framework to evaluate demographic characteristics such as gender, community, ethnicity, and race in exposure to crime (Hanson et al., 2000). Han and co-authors study and evaluate the relationship between use of local parks in low-income urban neighborhoods and the crime rates in Los Angeles during a 2-year study period (2013–

2015) (Han et al., 2018). Burley investigates the impact of Portland's green infrastructure initiative on neighborhood violence (Burley, 2018). Beland and Brent investigate the association of high traffic jam and crime to predict traffic's psychological expenses (Beland and Brent, 2018).

With respect to crime type-based research, RAND corporation studies the impact of business growth districts in Los Angeles on reported violent crime and youth violence (Rand, 2009). Ridgeway and MacDonal studied the effect of rail transportation on crime in neighborhoods near transit stations (Ridgeway and MacDonald, 2017). Levine and Wachs study bus incident crimes in Los Angeles (Levine and Wachs, 1986). Haberman study the extent to which hot spots of various crime types overlapped spatially (Haberman, 2017).

With respect to methodological crime research, Anderson and co-authors used geographical zoning on crime using over two hundred blocks chosen in eight various relatively high crime neighborhoods in Los Angeles (Anderson et al., 2013). Tahani and co-authors used spatial statistics find the criminal hotspots (Almanie et al., 2015). Also, Wand and co-authors use a deep learning spatio-temporal estimator, ST-ResNet to collectively approximate crime distribution over the Los Angeles area (Wang et al., 2017).

Based on the critical evaluation of the literature, there is a gap in using spatial-demographic information to study the crime in LA city and predict the crime rate. To address this gap, in our approach, we integrate two aspects of the research on crime, including criminal/victim features-based and methodological crime research over the Los Angeles city on the data set which is described in the Section 5.3.

5.3 Data Set Used for Crime Rate Prediction in Los Angeles City

5.3.1. Describing the data, features, and characteristics

Our crime dataset includes 12 month (January 210 to December 2010) for each of the 180 census tracts in Los Angeles.

Our dependent variable y_i is defined as the number of crime in census tract i($i = 1,\ldots,138$) for a total of 513 individual observations. In addition, in order to account for heterogeneity across census tracts, we collected data from the Census 2000 (US Census Bureau and Social Explorer Tables) on the following 55 socio-economic variables: number of people in each census tract (tot_pop), number of white people in each census tract (white), number of black people in each census tract (black), number of people in each census tract (native), number of people in each census tract (asian), number of no Hispanic Latin people in each census tract (nohisl), number of Hispanic Latin people in each census tract (hislat), number of male people in each census tract (male), number of female people in each census tract (female), number of male people in each census tract under the age of 5 (male_under 5), number of female people in each census tract under the age of 5 (female_under 5), number of male people in each census tract between 10 and 14 years old (male10-14), number of female people in each census tract between 10 and 14 years old (female10-14), number of male people in each census tract between 15 and 17 years old (male15-17), number of female people in each census tract between 15 and 17 years old (female15-17), number of male people in each census tract between 18 and 19 years old (male18-19), number of female people in each census tract between 18 and 19 years old (female18-19), number of male people in each census tract who

are 20 years old (male20), number of female people in each census tract who are 20 years old (female20), number of male people in each census tract between 22 and 24 years old (male22-24), number of female people in each census tract between 22 and 24 years old (female22-24), number of male people in each census tract between 25 and 29 years old (male25-29), number of female people in each census tract between 25 and 29 years old (female25-29), number of male people in each census tract between 30 and 34 years old (male30-34), number of female people in each census tract between 30 and 34 years old (male30-34), number of male people in each census tract between 35 and 39 years old (male35-39), number of female people in each census tract between 35 and 39 years old (female35-39), and number male and female between 40-44, 45-49, 50-54, 55-59, 60-61, 62-64, 65-66, 67-69, 70-74, 75-79, 80-84, 85and up.

In our descriptive analysis, we combined the age-related variables into 3 different age groups including under 17, between 18 and 39, and over 40 years old for male and female separately. So, instead of dealing with 55 variables we are dealing with 15 variables.

5.4 Methods Used for Crime Rate Prediction and Different Spatial Analysis

To create a spatial surrogate model for the dataset under study, some spatial statistics methods can be used. The most important ones are studied in this section. For example, in Section 4.4.1., point pattern analysis is explained as the most important spatial statistics technique while variance-mean ratio technique is discussed in Section 5.4.2. Also, in Section 5.4.3., distance-based point pattern measures are explained. Nearest neighbor distances method is explained in Section 5.4.4 while implementing the G, F, and K functions over the data set,

in sections 5.4.5., 5.4.6., respectively followed by the interpretation of the results in Section 5.4.8.

5.4.1. Point pattern analysis method to spatially analysis the location under study

Using point pattern analysis, we investigate the spatial organization of points in often two-dimensional space. We also want to identify any cluster structure exists in the spatial data. The hypothesize in point pattern analysis approach is that if any pattern observed, it has been produced by a particular spatial process, including independent random process (IRP) or complete spatial randomness (CSR). To shed more light on how point pattern analysis works, we are exploring the probability of the observed pattern being resulted by either IRP or CSR. Therefore, the spatial statistics is utilized to test this hypothesis.

If the variable under study, which is a crime distribution, follows a Poisson distribution, the number of points in spatial unit is a randomly distributed event or follows IRP or CSR process. This behavior is due to one of the critical characteristics of the Poisson distribution which is its mean and variance being equal to 1. In other words, if the variance mean ratio (VMR) of point pattern is near to 1, it means that the studied point pattern is randomly distributed or in other words, follows IRP or CSR. Accordingly, we can interpret the p-value as how probable is for the point pattern under study to have IRP or CSR. Thus, we cannot accept the null hypothesis if the p-value is smaller than 0.05.

5.4.2. Variance-mean ratio (VMR) analysis of crime rate prediction in Los Angeles city

Variance-mean ratio can be used to interpret the point pattern. There are two main scenarios, including the ratio less than or greater than 1. The distribution of the spatial units in the point pattern (here counties of the Los Angeles) is not homogenic when the VMR is greater than 1. In other words, there is more claustration in the point pattern what is expected if IRP or CSR exists. In contrary, the distribution of the spatial units in the point pattern (here counties of the Los Angeles) is homogenic when the VMR is less than 1. In other words, there is less claustration in the point pattern what is expected if IRP or CSR exists.

5.4.3. Distance-based point pattern measures

Another way of looking into the claustration level of point patterns is to use the distance-based point pattern measures. We can use the distances between points to determine if the point pattern does not follow the Poisson distribution and therefore is more clustered than expected.

5.4.4. Nearest neighbor distances

The nearest neighbor distance shows the average distance from each point in the area under study to its nearest point. By investigating the nearest neighbor distances, we can discover clustering or separation patterns within a point pattern. To measure the nearest neighbor distances, we can use different distance measures including the Euclidean distance as shown in Equation 5.1 as the distance between two events/points p_i and p_j as

$$Euclidean\ Distance\ (p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad \text{Equation 5.1}$$

Then, the average distance for each point is calculated using Equation 5.2.

$$Average\ Distance\ (p_i) = \frac{1}{n}\sum_{i=1}^{n} \min Distance\ (p_i)$$ Equation 5.2

Where min Distance (p_i) is the minimum of the distances between p_i and all other points in its neighborhood. Then, we can derive the expected value of the average nearest neighbor distance as Clark and Evans did in 1954 when the point patterns follow ICP or CSR as described in Equation 5.3 and Equation 5.4, where α is the intensity of the process (Clark and Evans, 1954).

$$Exp(Euclidean\ Distance\ ) = \frac{1}{2\sqrt{\alpha}}$$ Equation 5.3

Afterwards, we can use the "R" statistic as the ratio of the observed average nearest neighbor distance to the expected distance as shown in Equation 5.4.

$$R = 2 * \sqrt{\alpha} * \ Average\ Distance\ (p_i)$$ Equation 5.4

In our crime prediction example, the distribution of the spatial units in the point pattern (here counties of the Los Angeles) is not homogenic when R is much less than 1. In other words, there is more claustration in the point pattern what is expected if IRP or CSR exists. In contrary, the distribution of the spatial units in the point pattern (here counties of the Los Angeles) is homogenic when R is much greater than 1. In other words, there is less claustration in the point pattern what is expected if IRP or CSR exists.

5.4.5. Implementing the G function over the data set

The G function is the modified version of the nearest neighbor method and explains the cumulative frequency distribution of the nearest-neighbor distances. For a given distance *dis*, *G(dis)* gives the proportion of all nearest-neighbor distances that are less than *dis* as shown in Equation 5.5:

$$G(dis) = \frac{1}{n} * \ number\ of\ (Average\ Distance\ (p_i) < dis) \qquad \text{Equation 5.5}$$

5.4.6. Implementing the F function over the data set

F function, as shown in Equation 5.6, measures the distances between two points in the pattern and a set of random points in the location area under study. Assuming $x_1, x_2, \dots, x_n$ as a randomly chosen geographical points, and min $Distance\ (p_i, P)$ as the minimum distance from p_i to any geographical point in the point pattern.

$$F(dis) = \frac{1}{n} * \ number\ of\ (Average\ Distance\ (p_i, P) < dis) \qquad \text{Equation 5.6}$$

5.4.7. Implementing the K Function over the data set

The K function is created to overcome the drawbacks in F and G functions as they ignore so much information including the distance between a point and its non-nearest points. In order to measure the *K(dis)*, we put a circle of radius *r* centered on each point p_i in the point pattern. *C* (p_i, *r)* represents the circle. We count the number of other events inside each circle of radius d as shown Equation 5.7:

$$K(dis) = \frac{1}{n\alpha} \sum_{i=1}^{n} number\ of\ other\ points\ inside\ each\ circle\ of\ radius\ r\ (C\ (p_i, d)) \qquad \text{Equation 5.7}$$

As shown in Equation 5.7, the value of *K(dis)* is the average count for all points, and it is between 0 and 1 since it is scaled by the general intensity of the point patterns.

5.4.8. Interpretation of the results of hypothesis testing for F, G, K

When we are using the visualizations of these functions, we need to know under what circumstances each of these functions helps us to determine the level of claustration and separation in the point pattern. For example, G values are higher than expected when G curve is above expected line. Thus, the crime data over the spatial units is clustered while in contrary

random points are far from points or in other words, they are clustered when F curve is below the expected line. On the hand, we cannot interpret K function easily and in a straightforward way. However, generally it shows more linear behavior for an evenly spaced point pattern.

5.5 Spatial Prediction Model

Our goal is to predict crime using different census data. In order to achieve this goal, we followed three steps as shown in 5.1.

1) We first investigate which covariates might we include in the regression model. First a regular spatial regression model is fitted to the census data to predict the crime rate based on demographic data.

2) Afterwards, by conducting the Moran's I test, we justified that there is a spatial autocorrelation between the crime rate and demographic information of the Los Angeles census tracts.

3) We created a geographically weighted regression model to predict the crime rate using demographic data of Los Angeles.

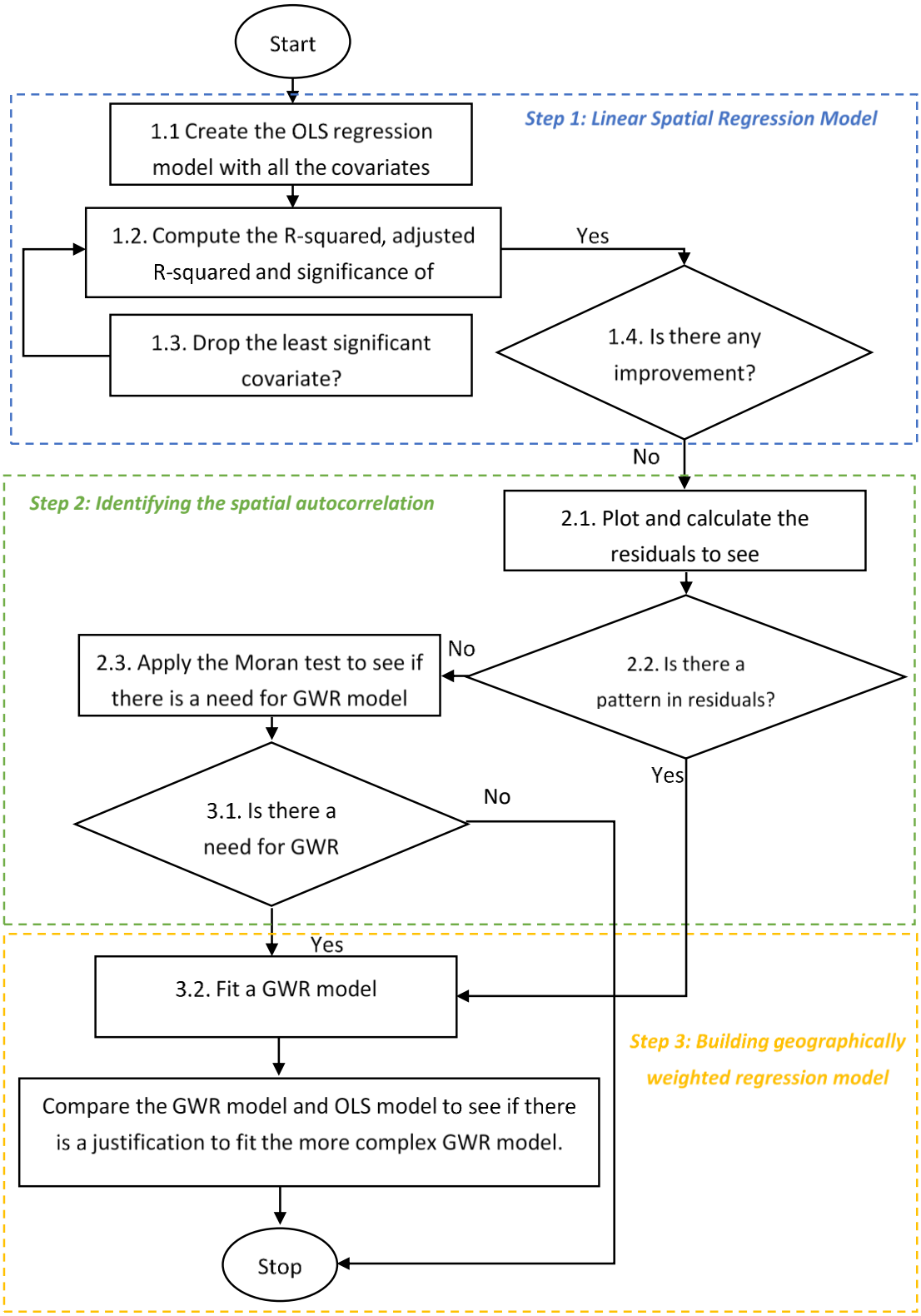


Figure 5.1. Spatial prediction model.

5.5.1. Step 1: Linear spatial regression model

In Step 1.1., we built a spatial linear regression model using all the attributes as predictors. Then, in Step 1.2., we conducted a F-statistic test and calculated the R-squared and the adjusted R-squared. In Step 1.3., we omitted the insignificant predictors from the model to make it as simple and accurate as possible. We check any improvements in terms of accuracy and the simplicity of the model.

5.5.2. Step 2: Identifying the spatial autocorrelation

In Step 2.1., we plot the choropleth[12] of residuals of the best fitted linear model in Step 1. If there is a meaningful spatial pattern in the choropleth and the global model "mis-specified", there is justification for fitting a geographically weighed regression model. If we do not find any noticeable pattern in the residual choropleth, we move in Step 2.3. In Step 2.3., we apply a Moran's I test to see if there is a need to fit the GWR model using Equation 5.8.

$$I = \left[\frac{m}{\sum_{j=1}^{m}(y_j - \bar{y})^2}\right] * \left[\frac{\sum_{j=1}^{m}\sum_{i=1}^{n} w_{ji}(y_j - \bar{y})(y_i - \bar{y})}{\sum_{j=1}^{m}\sum_{i=1}^{n} w_{ji}}\right] \qquad \text{Equation 5.8}$$

Where the first argument of the equation shows the total number of the point patterns divided by the overall variance of the data set. The nominator of the second argument shows the covariance term and subscript I and j represent different areal units while the denominator of the second argument is to show the normalization using the overall spatial weights in the point pattern. If there is a considerable reason to have a spatial correlation, we fit the GWR model in Step 3.2. To compute the Moran's I value, we use K nearest neighbors for spatial

[12] A type of statistical thematic map that uses intensity of color to correspond with an aggregate summary of a geographic characteristic within spatial enumeration units, such as population density or per-capita income.

weights. The function returns a matrix with the indices of points belonging to the set of the k nearest neighbors of each other, where K represents the number of neighbors to each county. So, we try different K values to find the best K value which leads to the least p_value which indicates significant spatial autocorrelation.

5.5.3. Step 3: building geographically weighted regression model

After conducting the Moran's I test as described in Step 2.3 and concluding that there is meaningful spatial autocorrelation between the number of crime in each county and the spatial-demographic attributes, we build a geographical weighted regression model using the Gaussian kernel following these steps:

1. Set a random bandwidth as the initial value for the number of nearest neighbors of each spatial unit. The bandwidth is calculated using Equation 5.9.

$$\text{Gaussian: weight} = \exp(-0.5 * (\text{vdist}/\text{bw})\text{^}2) \qquad \text{Equation 5.9}$$

   Where *vdist* stands for the distance matrix between the spatial units and *bw* stands for the bandwidth.
2. Because we find the most appropriate bandwidth, it does not matter which value to choose as the initial value.
3. Create the basic geographical weighted model based on the initial bandwidth.
4. Improve the GWR model by changing the bandwidth value. To find the best bandwidth value, we calculate the Akaike's Information Criterion (AIC). AIC is generally calculated using Equation 5.10:

$$AIC = -2(log - likelihood) + 2K \qquad \text{Equation 5.10}$$

Where, K stands for the number of model parameters (the number of variables in the model plus the intercept). Also, log-likelihood is a measure of model fit. The lower the AIC value, the better the fit. It is usually obtained from statistical output.

5.6 Results and Discussion of Spatial Prediction Model

5.6.1. Visualization of the distribution of the crime data over the Los Angeles city

First, we visualize the distribution of the crime data over the Los Angeles city. As can be seen in the Figure 5.2, there is a pattern in the distribution of the crime over the region of study.

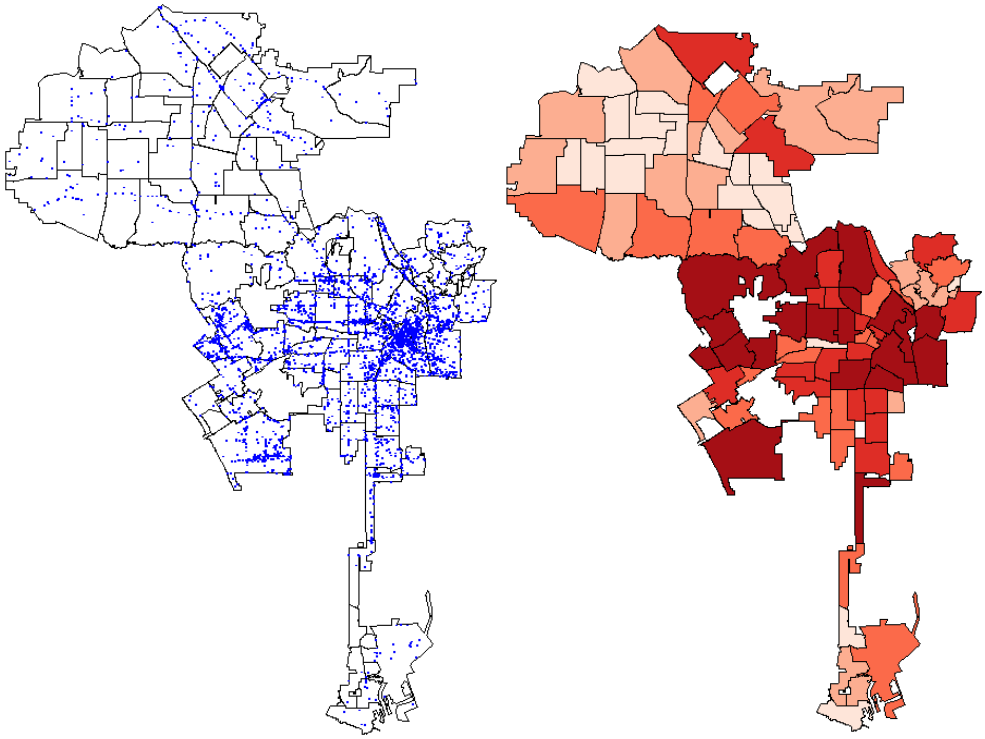
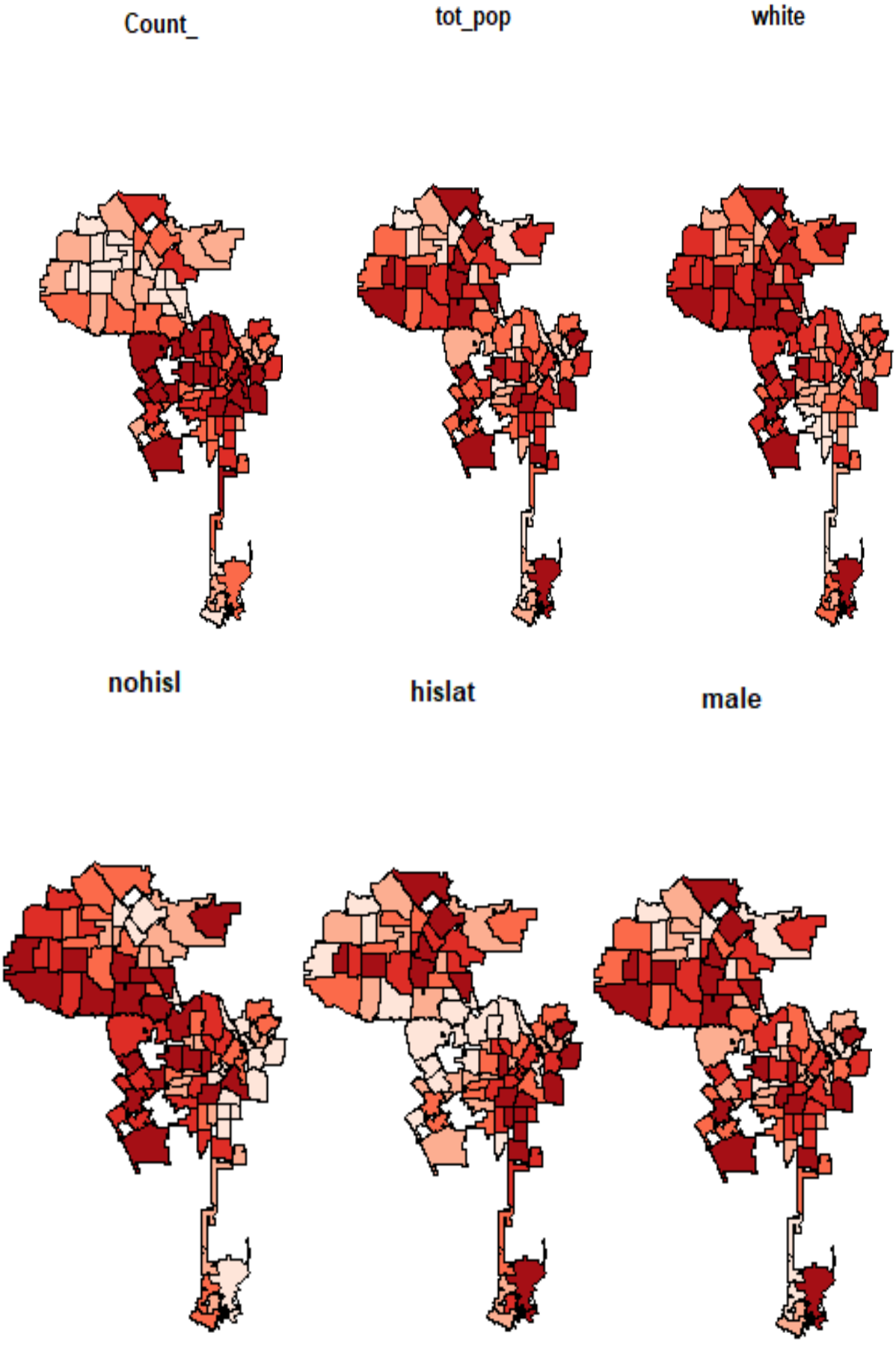


Figure 5.2. Distribution of the crime over the Los Angeles census tracts. (produced by author)
Note: Darker colors show higher crime rate

Now, before creating the crime prediction model we plot the choropleth of the covariates. As can be seen in Figure 5.2, almost all the covariates show a sort of pattern like the crime rate. So, we decide to create the prediction model using all the covariates.


Count_
tot_pop
white
nohisl
hislat
male

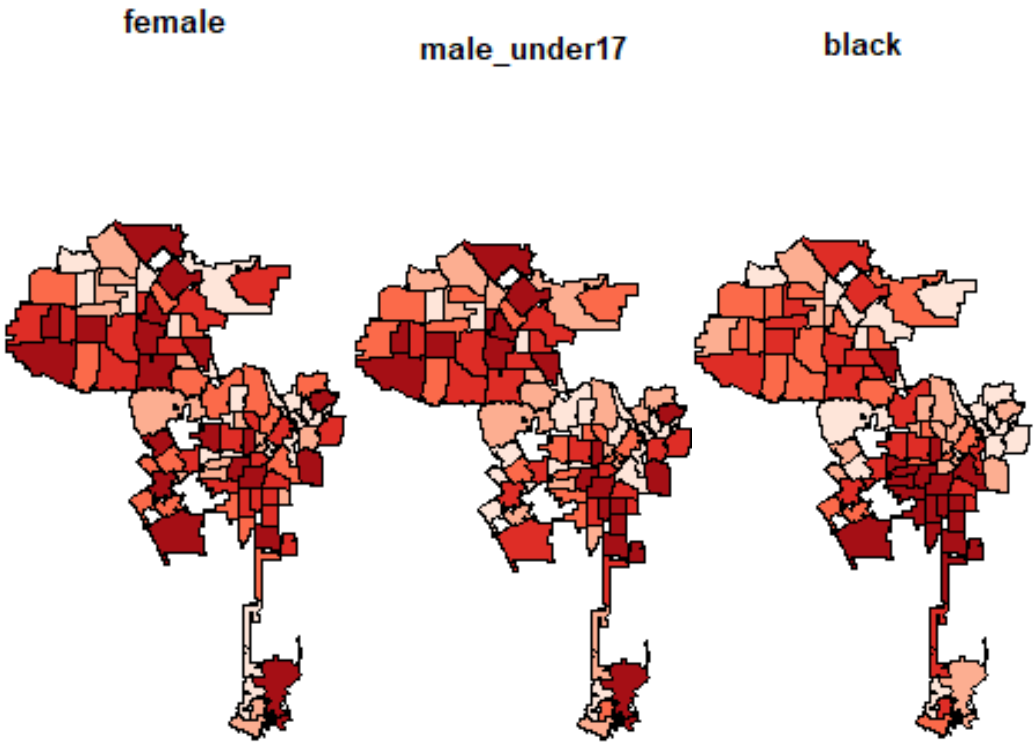


Figure 5.3. Visualization of the patterns in each predictor/covariate over the LA Census Tracts. Note: Darker colors show higher value for the corresponding variable (produced by author).

5.6.2. Results and discussion of fitting OLS linear regression

Based on the preliminary visulization of the covariates and the response variable, we decided to create the regression model with all the covariates. So, as shown in Table 5.1, we have built the prediction model based on all the covariates and then try to eliminate the covariates which are not statistically significant in predicting the crime rate.

Table 5.1. Results of fitting the spatial linear regression model.

	Predictors	**R^2**	**Adjusted R^2**	**P-value**
Model 1	pointpatNew$tot_pop+pointpatNew$white+pointpatNew$black+pointpatNew$native+pointpatNew$asian+pointpatNew$nohisl+pointpatNew$hislat+pointpatNew$male+pointpatNew$female+pointpatNew$male_under17+pointpatNew$male18to39+pointpatNew$male_over40+pointpatNew$female_under17+pointpatNew$female18to39 +pointpatNew$female_over40	0.477	0.409	2.4e-08
Model 2	All the variable in Model 1 except: pointpatNew$hislat; pointpatNew$female; pointpatNew$male_over40; pointpatNew$female_over40	0.477	0.409	2.4e-08

	Predictors	R^2	Adjusted R^2	P-value
Model 3	All the variable in Model 2 except: pointpatNew$female_under17; pointpatNew$native	0.468	0.413	5.2e-09
Model 4	All the variable in Model 3 except: pointpatNew$male_under17; pointpatNew$white	0.389	0.341	1.4e-07

We built a spatial linear regression model using all the attributes as predictors. Then, we conducted a F-statistic test and calculated the R-squared and the adjusted R-squared. We omitted the insignificant predictors from the model to make it as simple and accurate as possible. As shown in Table 5.1, the Model 3 is the simplest accurate model. Based on the Model 3 we plotted the choropleth of its residuals as described in Step 2.1. As shown in Figure 5.4. Since it is difficult to tell if there is spatial autocorrelation in these residuals by eye, as described in Step 2.2., we need to measure it before going further. For this purpose, we conducted the Moran's I test as described in Step 2.3.

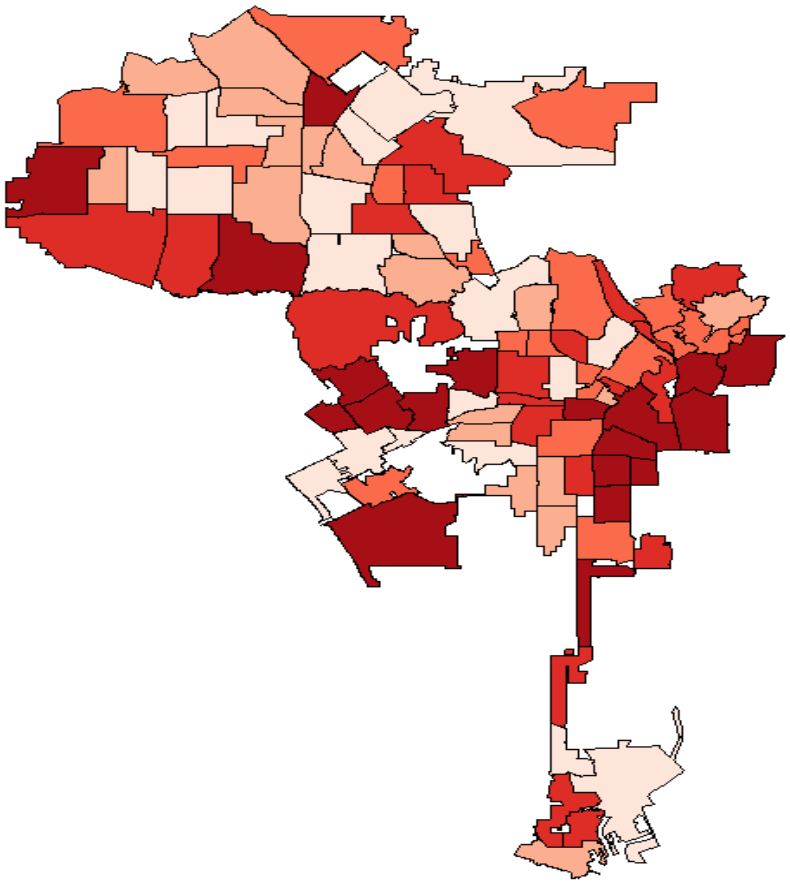


Figure 5.4. Residual pattern of the linear Model 3. (Produced by author)

In order to calculate the Moran's I value, we use K nearest neighbors for spatial weights. The function returns a matrix with the indices of points belonging to the set of the k nearest neighbors of each other, where K represents the number of neighbors to each county. So, we try different K values to find the best K value which leads to the least p_value which indicates significant spatial autocorrelation. Table 5.2 shows the results of the Moran's I test through K nearest neighbors' method.

Table 5.2. Results of Moran's I test using K nearest neighbors' method.

	Number of neighbors	**Moran's I statistic**	**P-value**
Moran's I test 1	K =1	0.10443135	0.1541
Moran's I test 2	K=2	0.140642300	0.0398
Moran's I test 3	**K=3**	**0.17069461**	**0.005961**
Moran's I test 3	K=4	0.125052476	0.01586

As shown in Table 5.2, we have a significant (but small) positive spatial autocorrelation in the residuals from the global regression model. So, we have some justification for fitting a GWR model.

5.6.3. Result and discussion of fitting the GWR model

We build a geographical weighted regression model using the Gaussian kernel and a bandwidth of 50km as the initial point. Then, using Equation 5.9 and Equation 5.10, we calculate the AIC value for different bandwidth values. As shown in Figure 5.5, the bandwidth value of 9.14171 has the highest AIC value of 1141.18. So, we set the bandwidth value and create the GWR model based on that.

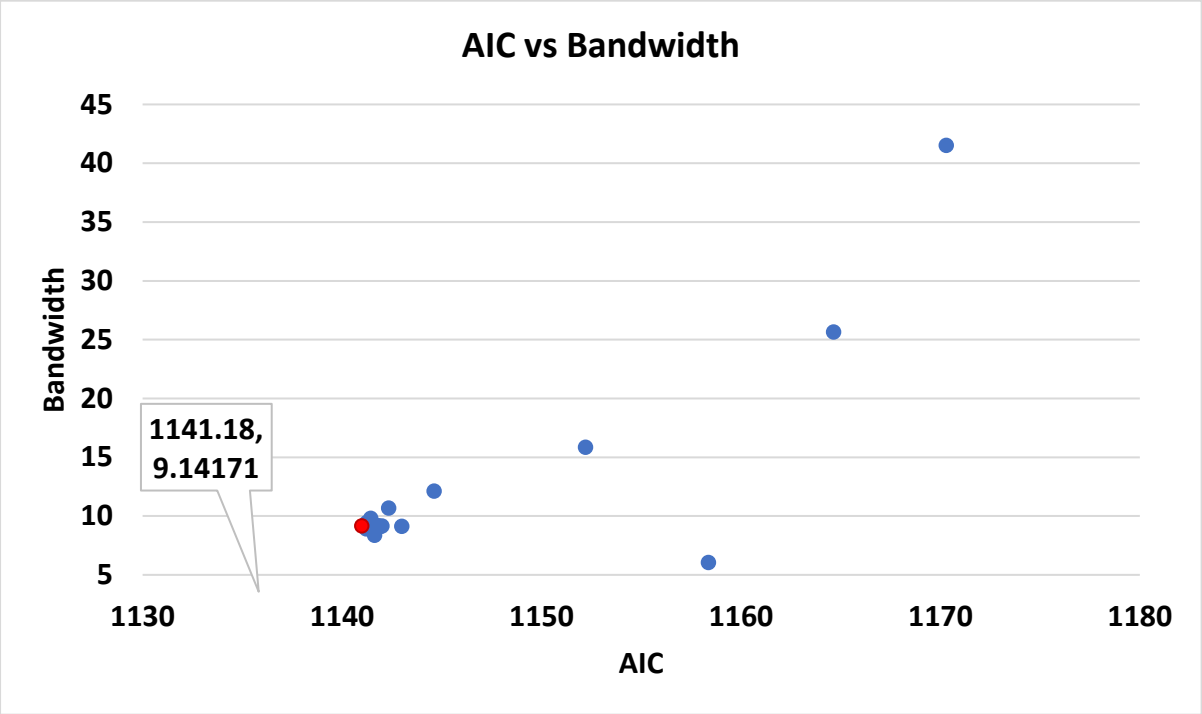


Figure 5.5. The AIC values for different bandwidths

Now, using the best bandwidth value of 9.14171, we build the GWR model.

5.6.4. Results and discussion of comparison between OLS and GWR models.

To see if fitting GWR model improves the previously found results of OLS model and if there a justification to fit this more complex model, we calculate the F-statistics, R-squared and Adjusted R-Squared for both OLS and GWR models.

Table 5.3. Comparison between OLS and GWR models.

	Predictors	**R^2**	**Adjusted R^2**	**P-value**
OLS Model	pointpatNew$Count_~pointpatNew$tot_pop+pointpatNew$white+pointpatNew$black+pointpatNew$asian+pointpatNew$nohisl+pointpatNew$male+pointpatNew$male_under17+pointpatNew$male18to39+pointpatNew$female18to39	0.4679	0.4128	5.154e-09
GWR Model	pointpatNew$Count_~pointpatNew$tot_pop+pointpatNew$white+pointpatNew$black+pointpatNew$asian+pointpatNew$nohisl+pointpatNew$male+pointpatNew$male_under17+pointpatNew$male18to39+pointpatNew$female18to39	**0.8066492**	**0.6860462**	**5.154e-09**

Based on the comparison results which are summarized in Table 5.3, we find huge improvement in the created model, and we conclude that it is justified to use the more complex GWR model.

5.6.5. Results and discussion of Clark-Evans R statistics analysis

We use the Clark-Evans R statistic as the ratio of the observed mean nearest neighbor distance to that expected for a Poisson point process of the same intensity. Using Clark-Evans R statistic, we find for which point pattern should R be greater than 1 (less clustered)? Less than 1 (more clustered)? About 1 (not clustered)? The R value for a two-sided Clark-Evans test with no edge correction equals 0.90323 with low significance (p-value = 0.06826). We implemented the Donnelly correction and Monte Carlo test based on 999 simulations of CSR and the R value became 0.86401, with the significance of p-value = 0.014.

Based on the mean nearest neighbor distance and Clark-Evans test, R value is less than 1 but not so much. So, we implemented G, F and K statistics tests calculating the Monte Carlo envelopes for the point pattern which the results are shown in Figure 5.6.

5.6.6. Results and discussion of G, F, and K statistics analysis

Based on the results shown in Figure 5.6, since G curve is above expected line, G values are higher than expected and so the crime data over the spatial units is clustered. Also, since F curve is below expected line, random points are further from events and so the crime data over the spatial units is clustered. In addition, K function is also showing that the crime data over the spatial units is clustered.

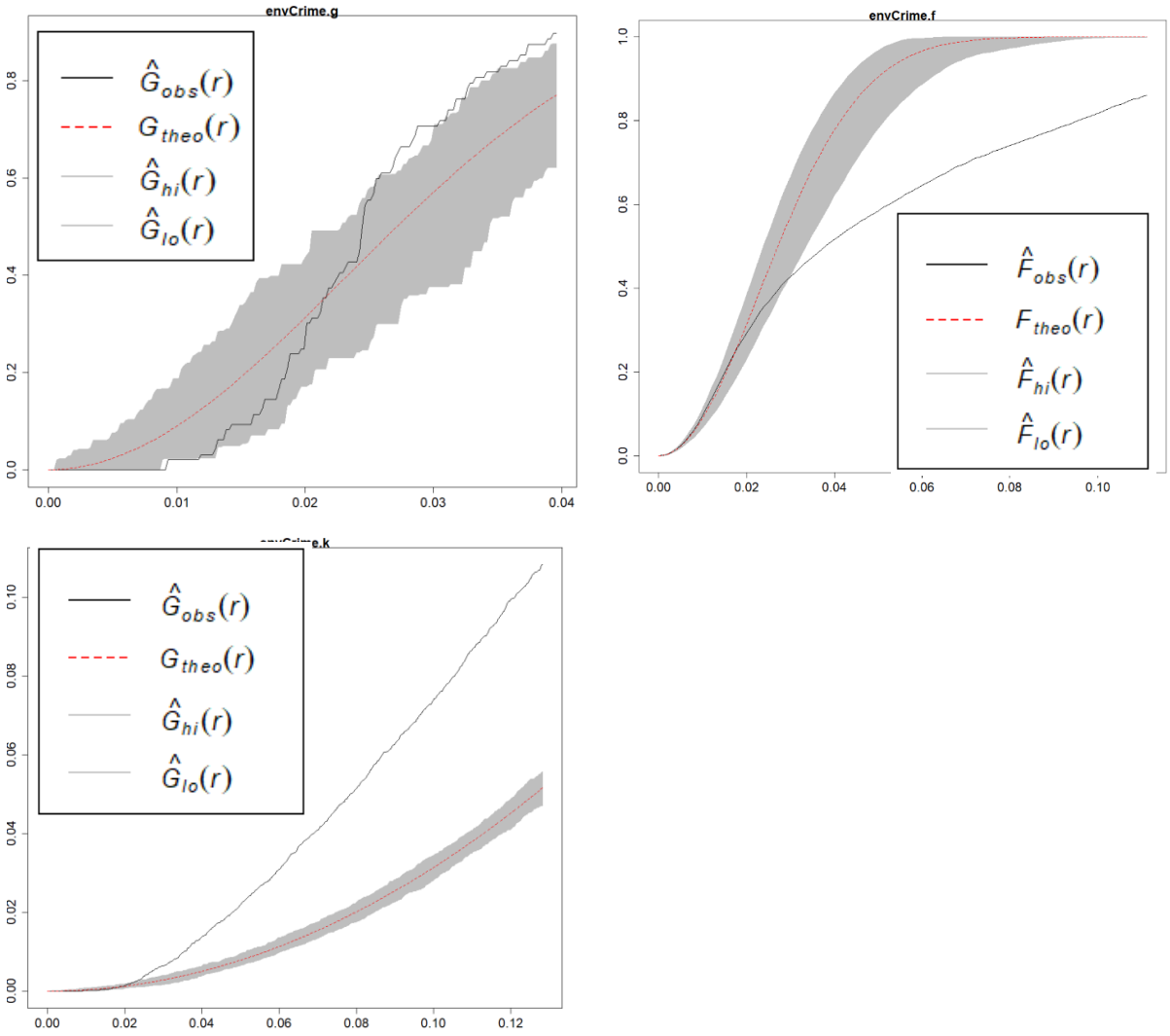


Figure 5.6. Results of G, F and K statistics tests.

5.7 On Verification and Validation – Empirical Structural and Performance Validation

Empirical structural validation involves accepting the appropriateness of the example problems used to verify the performance of the framework and the method. The spatial surrogate models are first tested using an example problem in Chapter 5. In the studied example problem, the possibility of integration of spatial statistics with surrogate modeling is considered. Using the framework and method, crime rate in Los Angeles city is predicted using socio-

demographic and geographical data of the census blocks of the city. The example thus is appropriate to demonstrate the utility of the framework and method as it involves spatial features, and it is a computationally expensive problem that needs systematic problem formulation and uncertainty and complexity management to design the entire system. The example is appropriate as the example supports in demonstrating the utility of method and framework in carrying out spatial prediction. The example is further improved and expanded to the comprehensive problem on two-echelon, green supply chain design in Chapter 8. Using the comprehensive example problem in Chapter 8, the utility of the framework and the method is tested for the supply chain demand prediction. In Chapter 5, Gap 5, which is on integrating surrogate models and spatial statistics is posed after a critical evaluation of the literature. To address this gap, Research Question 5 is posed as "what is the appropriate SM to use when the data includes spatial variables as predictors?". Then to answer this question, we hypothesized that through replacing the design of experiments with the geographically weighted correlation analysis, we find the surrogate model which is useable with spatial variables. By testing this hypothesis through creation of spatial surrogate models and testing it on crime detection problem in Los Angeles city, we achieved the key outcome of using the spatial statistics and particularly the concept of the geographically weighted regression in surrogate modeling to build spatial surrogate models.

5.8 Closing Remarks on Predicting Crime Rate using Spatial Surrogate Models

Our contribution is important at three perspectives: computation, criminology, and law enforcement. From a computational perspective, we demonstrate the feasibility of using GWR

model for spatial-demographic data. It also allows for evaluation of a wide range of additional statistics of empirical relevance such as mean nearest neighbor distance, G, F, and K functions for point pattern analysis. From a criminological perspective, our results relative to the impact of socio-economic covariates on crime rate largely support prevailing conjectures in the literature. Moreover, they give us enough reason to not accept the 'random distribution' hypothesis of the crimes and enable us to use global geographically weighted regression model. This implies that there are clusters in the crime rate in Los Angeles city and we showed these clusters. Last but not least, the computation of OLS as well as GWR models enable us to predict future crime rate and prevent the spatial diffusion of the crimes by focusing on high crime rate areas. In combination with the forecast statistics we believe that our model can play a helpful role in the effective allocation of rare law enforcement resources, in line with but more detailed than in the pioneering results of (Cohen and Gorr, 2005; Cohen et al., 2007).

CHAPTER 6 CREATING MATHEMATICS FOR BUILDING ENSEMBLE OF SURROGATE MODELS

In this chapter, Hypothesis 2, which is using a weighted average of individual surrogate models and minimum overall cross validation error, we can build such ensemble of surrogate models which are relatively less computationally complex and more accurate, is tested using a hot rod rolling test problem. In this example problem, two proposed methods are tested: first, a framework to classify and select the surrogate models based on time, size and accuracy criteria; second, a method to build an ensemble of surrogates (EoS), which is both accurate and less computationally intensive. The surrogate models are chosen and combined systematically using well-established theoretical and empirical models and simulation experiments (finite-element based). Using the proposed framework and method, the integrated surrogate model selection and ensemble creation process is carried out in an integrated approach. The example thus is appropriate to demonstrate the utility of the framework and method as it involves complex information flow across manufacturing stages that needs approximation through surrogate modeling and the simulation data is sparse. The contribution offered in this chapter is to propose a method based on cross-validation. to find an EoS which is created by the least possible number of data points. The resulting ensemble surrogate has higher accuracy than each individual surrogate and is less computationally intensive. To achieve this ensemble surrogate, we compare it with individual surrogate models based on computation time, size, and desired accuracy. For this purpose, we use root mean square error (RMSE) as the accuracy measure, time of simulation as the computation performance measure, and the number of data points as

the dimension measure. In summary, we find that (1) it is effective to use cross-validation to study the impact of the size of the sample data set; (2) the highest accuracy with least required data and less computation time is achievable using the right number of samples; and (3) an example of surrogates is relatively insensitive to the size of the sample data or number of data points.

Summary of Building Ensemble of Surrogate Models

In engineering design, surrogate models are often used instead of costly computer simulations. Typically, a single surrogate model is selected based on previous experience. We observe, based on an analysis of the published literature, that fitting an ensemble of surrogates based on cross-validation errors is more accurate but requires more computational time. In this chapter, we propose a method to build an ensemble of surrogates that is both accurate and less computationally expensive. In the proposed method, the ensemble of surrogates is a weighted average surrogate of response surface models, Kriging, and radial basis functions based on overall cross validation error. We demonstrate that created ensemble of surrogates is accurate than individual surrogates even when fewer data points are used, so, computationally efficient with relatively insensitive predictions. We demonstrate the use of an ensemble of surrogates using hot rod rolling as an example. Finally, we include a rule-based template which can be used for other problems with similar requirements, e.g., the computational time, required accuracy, and the size of the data.

Glossary of Mathematics of Building Ensemble of Surrogate Models

CFD Computational Fluid Dynamics

DOE	Design of Experiments
EA	Evolutionary Algorithms
GSME	Generalized Mean Square Error
NSGA-II	Non-dominated Sorting Genetic Algorithm II
POF	Pareto Optimal Front
PRESS	Predicted Residual Error Sum of Squares
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
RMSE	Root Mean Square Error
RSM	Response Surface Models
SE	Squared Error
WAS	Weighted Average Surrogate
EoS	Ensemble of Surrogates

6.1 Frame of Reference on Building Ensemble of Surrogate Models

Computer simulations are commonly used to replace experiments with physical models. Often these simulations are computationally expensive. However, many model-based engineering design problems require numerous simulations to reach an acceptable solution. This can be computationally prohibitive.

Often a single surrogate model – or metamodel – is used to replace a detailed simulation in design problems which require repeated calculations. This surrogate is obtained using information derived from the physical model. Many types of surrogate models have been

proposed. Here, we demonstrate the advantages of computing ensemble of surrogates from a single set of data and then averaging these surrogates to make use of the good features of each type of surrogate. We term these assemblies of surrogates, ensembles of surrogates. The ensemble of surrogates is built using the weighted average of different individual surrogates. These weights can be calculated randomly or in a systematic way. For instance, (Viana and Haftka, 2008) use a systematic weighted average surrogate (WAS) to utilize the advantage of *n* surrogates to cancel the errors in estimation by appropriate weight selection in the linear mix of the models. They use an ensemble of metamodels to minimize the RMSE in surrogate modeling. They discuss using the lowest predicted residual error sum of squares (PRESS) solution or just an average surrogate when an individual surrogate is required. They also propose the optimization of the integrated square error (SE) as an approach to calculate the weights of the WAS. They found that it is worthwhile to create a broad set of various surrogates and after that apply PRESS as the selection criterion.

Cross-validation is utilized broadly to allocate the weights to individual surrogates in building an ensemble of surrogates in a systematic way. (Viana et al., 2009) use cross validation to approximate the necessary safety border for a particular favorite conservativeness degree (safe approximations percentage). They also check how well they can reduce the loss of accuracy caused by a conservative estimator[13] by choosing among alternative surrogate models. They show that cross-validation enables to choose the best conservative surrogate model with the lowest accuracy loss. Also, they found that it is efficient in determining safety is effective

[13] The estimator is the predicted residual error sum of squares (PRESS) and obtained by dividing the *N* data points into *k* subsets in cross-validation process. It is estimated by using a subset of points in building the surrogate and computing the errors at these left out points. Then, this process is replicated with various sets of left-out points to obtain PRESS which is statistically significant.

for selecting the safety edge. Also, Viana and coauthors compare using the lowest PRESS with a weighted surrogate when an individual surrogate is required.

They propose optimizing the incorporated squared error (SE) as an approach to calculate the weights of the WAS model. They find that it is better to create a big set of various surrogate models and choose the best based on the PRESS and that the error of cross-validation provides a great approximation of the RMSE if enough data points are used. However, in high dimensions, the advantages of using the cross-validation error and weighted surrogates are decreased considerably. Also, (Goel et al., 2007) create a systematic heuristic process for computation of the weights as the PRESS weighted average surrogate (PWS). Using a combination of neural networks, (Bishop, 1995) create a systematic weighted average surrogate gained by estimating the covariance between surrogates from residuals at test or training datasets. Following Bishop's method, (Acar and Rais-Rohani, 2009) develop another approach to optimizing the mean square (MS) error.

As shown in Table 6.1, we critically evaluate the surrogate modeling literature based on the type of research, the number of the combined surrogates, weight assignment process, and comparison criteria.

Table 6.1. Critical evaluation of the ensemble of surrogates' literature.

Paper \ Features	Type of research (Theoretical/ Experimental)	Number of combined surrogates	Method of specifying weights for the ensemble surrogates (systematic/random)	Criteria used to compare the methods (time/size/accuracy)		
				Accuracy	Time	Size
(Mack et al., 2005)	Experimental	3	Random	*		
(Samad et al., 2006)	Theoretical	4	Random	*		
(Samad et al., 2006)	Experimental	3	Random	*		
(Goel et al., 2007)	Theoretical	3	Systematic	*		
(Lim et al., 2007)	Theoretical	4	Random	*		
(Samad et al., 2007)	Experimental	3	Systematic[14]	*		
(Viana and Haftka, 2008)	Theoretical	4	Systematic[15]	*		
(Viana et al., 2009)	Theoretical	4	Systematic[16]	*		
(Bhat et al., 2010)	Experimental	4	Random	*		

Paper \ Features	Type of research (Theoretical/ Experimental)	Number of combined surrogates	Method of specifying weights for the ensemble surrogates (systematic/random)	Criteria used to compare the methods (time/size/accuracy)		
				Accuracy	Time	Size
(Viana et al., 2010a)	Theoretical	4	Systematic	*		
(Arias-Montano et al., 2012a)	Theoretical	5	Systematic	*		
(Basudhar, 2012)	Theoretical	5	Random	*		
(Viana et al., 2013)	Theoretical	10	Systematic	*		
(Villanueva et al., 2013)	Theoretical	4	Systematic	*		
(Chaudhuri and Haftka, 2014)	Theoretical	4	Systematic	*		
(Chaudhuri and Haftka, 2014)	Experimental	4	Systematic	*		
(Acar, 2015)	Theoretical	4	Systematic	*		
(Adhav et al., 2015)	Experimental	3	Systematic	*		
(Badhurshah and Samad, 2015)	Theoretical	2	Systematic	*		
(Chaudhuri et al., 2015)	Theoretical	2	Systematic	*		
(Liu et al., 2015)	Theoretical	24	Systematic	*		
(Babaei and Pan, 2016)	Experimental	9	Systematic	*		
(Alizadeh et al., 2016b)	Experimental	4	Systematic	*		
(Bellary et al., 2016)	Experimental	2	Systematic	*		
(Beynaghi et al., 2016)	Theoretical	4	Systematic	*		
(Bhattacharjee et al., 2016)	Experimental	4	Random	*		
(Qiu et al., 2016)	Experimental	4	Random	*		
(Shankar Bhattacharjee et al., 2016)	Experimental	7	Random	*		
(Shi et al., 2016)	Theoretical	4	Systematic	*		
(Kaleibari et al., 2016)	Theoretical	4	Systematic	*		
(Wang et al., 2016)	Theoretical	4	Systematic	*		
(Bellary and Samad, 2017)	Theoretical	3	Systematic	*		
(Habib et al., 2017)	Theoretical	5	Systematic	*		
(Bhattacharjee et al., 2018)	Theoretical	4	Systematic	*		
(Zamani Sabzi et al., 2018)	Theoretical	4	Systematic	*		
(Ezhilsabareesh et al., 2018)	Experimental	6	Systematic	*		
(Lv et al., 2018)	Theoretical	5	Systematic	*		
(Song et al., 2018)	Theoretical	4	Systematic	*		
(Viana and Haftka, 2008)	Theoretical	10	Systematic	*		
(Song et al., 2018)	Theoretical	3	Systematic	*		
(Yin et al., 2018)	Theoretical	3	Systematic	*		

Experimental ensembles of surrogates calculated from a single set of data points can be used to overcome the weaknesses of every single type of surrogate. For instance, (Song et al., 2018) study the efficiency of using an ensemble of surrogates in improving accuracy and the robustness for several problems. Robustness is the ability of the model to have low fluctuation in accuracy in different situations (e.g., with low and high amount of data). They use an integrated ensemble surrogate model to: (i) filter out the individual models with low performance and retain the higher-performing ones using cross-validation errors; (ii) calculate the appropriate weighting for each surrogate model included in the ensemble based on the reference model and the approximated mean square error (MSE). (Xu and Zeger, 2001) use an

ensemble of surrogates and introduce two independent processes to highlight their advantages instead of individual surrogates. On the other hand, (Zhou et al., 2018b) study the drawbacks of compound and ensemble surrogates and their inadequacy based on quasi-concavity. (Samad et al., 2006) analyze the use of an ensemble of surrogates and performance approximation simultaneously.

Ensembles of surrogates are often used in the surrogate-assisted design. For instance, (Viana et al., 2013) utilizes an ensemble of surrogate modeling approach when adding more than one point in each optimization iteration. Existing global optimization algorithms may be revised to find multiple alternative designs, however, parallel computation is the key to increasing optimization efficiency (Chaudhuri and Haftka, 2014; Villanueva et al., 2013). (Bhattacharjee et al., 2018) implement an ensemble of surrogates assisted multiple-objective optimization for engineering design problems which are computationally costly. Also, (Chaudhuri and Haftka, 2014) use an ensemble of surrogates to compute Pareto optimal fronts (POFs). An ensemble of surrogates is utilized to decrease uncertainty in searching for an optimal point (Adhav et al., 2015). (Liu et al., 2015) utilize an ensemble of surrogates with a genetic algorithm (GA). (Badhurshah and Samad, 2015) find that using an ensemble of surrogate assisted optimization methods and computational fluid dynamics analysis, the optimality, efficiency of the optimization process, and the robustness of the optimum solutions can be improved. (Wang et al., 2016) use an ensemble of surrogates for global optimization to enhance the convergence ratio of an uncertainty predictor. (Samad et al., 2006) evaluate the performances of ensembles of surrogates in a turbomachinery blade-shape optimization. They use RSM, Kriging, RBNN and weighted average models to test shape optimization. They find that using an ensemble of surrogates through weighted averaged surrogates provides more robust estimation than single ones.

Some authors analyze the fidelity of the estimation functions modeling in surrogate-based optimization in engineering design. (Bellary and Samad, 2017) address this issue using the ensemble of surrogates to suggest estimations from alternate modeling schemes. Also, (Habib et al., 2017) use an ensemble of surrogates assisted optimization method and evaluate it at various levels of fidelity. (Yin et al., 2018) propose assembling an ensemble of surrogates by dividing the design space into several subspaces such that each is allocated a collection of optimized weights. (Acar, 2015) argues for giving greater importance to maximum error (MAXE) than root mean square error (RMSE) by assigning weights of the individual surrogates in the ensemble of surrogates. In this chapter, weights for the surrogate models in the ensemble of surrogates are selected to minimize the root mean square cross validation error (RMSE-CV) in a hope to minimize the original RMSE. Additionally, some studies are specifically focused on the ensemble of surrogates of just one type of metamodel. For instance, (Shi et al., 2016) introduce a combination of RBFs to determine the weights by solving a quadratic programming (QP) subproblem. The results show that an ensemble of multiple RBFs can remarkably enhance the modeling efficacy compared to single RBF models.

Many authors show the use of the ensemble of surrogates in evolutionary algorithms. (Lim et al., 2007) study the search efficiency of various surrogate modeling methods and the ensemble of surrogates in a memetic surrogate-based technique. (Bellary et al., 2016) use ensemble of surrogates integrated into a genetic algorithm (GA) to acquire POFs. They realize that the WAS ensemble of surrogates has better performance for both the goals than a single metamodel. Ensemble of surrogates and Non-dominated Sorting Genetic Algorithm II (NSGA-II) is used to optimize a simultaneous structure control design strategy (Bhat et al., 2010). They found that by introducing a new weighting approach as a frequency-dependent function it is possible to minimize closed-loop measures for optimal performance by searching over the design space encompassing the open-loop dynamic and controller variables while keeping

constraints. (Arias-Montano et al., 2012b) use ensembles of surrogates combined with an evolutionary method to obtain the benefit of their advantages for solving expensive multiple objective optimization problems.

(Bhattacharjee et al., 2016) introduce a multiobjective evolutionary algorithm embedded with different surrogates which are spatially distributed. They use a nondominated sorting genetic algorithm as the underlying optimizer. They extract the best features of different strategies and show that the multiple surrogates assisted multiobjective optimization with local surrogates with improved pre-selection offers better performance than individual surrogates. The same group, in another study, compares ensembles of surrogates in surrogate assisted multi-objective optimization algorithm (SAMO) with NSGA-II. They find that SAMO consistently outperforms NSGA-II (Bhattacharjee et al., 2016). (Lv et al., 2018) use an integrated framework of an ensemble of surrogates in particle swarm optimization (PSO), which includes inside and outside optimization loops. In the outside optimization loop, a PSO algorithm is used for both the sampling and the optimization approaches. In the inside optimization loop, an ensemble of surrogates assisted parallel optimization approach is implemented. They show that their framework can converge to a good solution for non-convex, multimodal and low-dimensional problems. use it to find the best location for speed bump installation using experimental design methodology. (Ezhilsabareesh et al., 2018) use an ensemble of surrogates-based multi-objective optimization approach using RMS, Kriging, RBNN, and NSGA-II. They find that among the surrogates, RSM delivers lower PRESS and has the best PRESS for all of the objectives. They also find that RMS produces the lowest RMS error after evaluation by NSGA-II.

In summary, building an ensemble of surrogates results in higher accuracy in many cases but it is more computationally intensive than using individual surrogates. In this chapter, we

address this gap in the published literature on creating a less computationally intensive ensemble of surrogates. Ensembles of surrogates and cross-validation as shown in Table 6.2, two types of studies have been done, namely, experimental, and theoretical. Also, four surrogates are often combined to build an ensemble of surrogates. Additionally, the weight assignment process has changed from a random to a systematic procedure. Finally, almost all studies use the only accuracy as the criteria for comparing the performance of the ensemble of surrogates with each individual surrogate model. However, more appropriate comparison criteria are computational time, size and accuracy and an understanding of a method for making trade-offs among these attributes (Alizadeh et al., 2020a). A qualitative description of the trade-offs among these three criteria is shown in (Alizadeh et al., 2020a).

Table 6.2. Trade-offs among three criteria.

Situation	**Detail trade-off**
If computational time is fixed	The larger the size (complexity) of the problem, the lower the accuracy
If the problem size is fixed	The higher the accuracy, the greater the computational time
If the desired accuracy is fixed	The larger the size (complexity) of the problem, the greater the computational time

The computational time represents the sum of the program running time which is used to construct the surrogate model and the simulation time used for sampling. So, in this work, the time is $T = T_{program} + T_{simulation}$. Also, here the problem size is defined as the amount of the sample data required, and the desired accuracy is evaluated by the deviation of the predicted response of the surrogates from the response of the simulation. The interactions among the three criteria are shown in Figure 6.1.

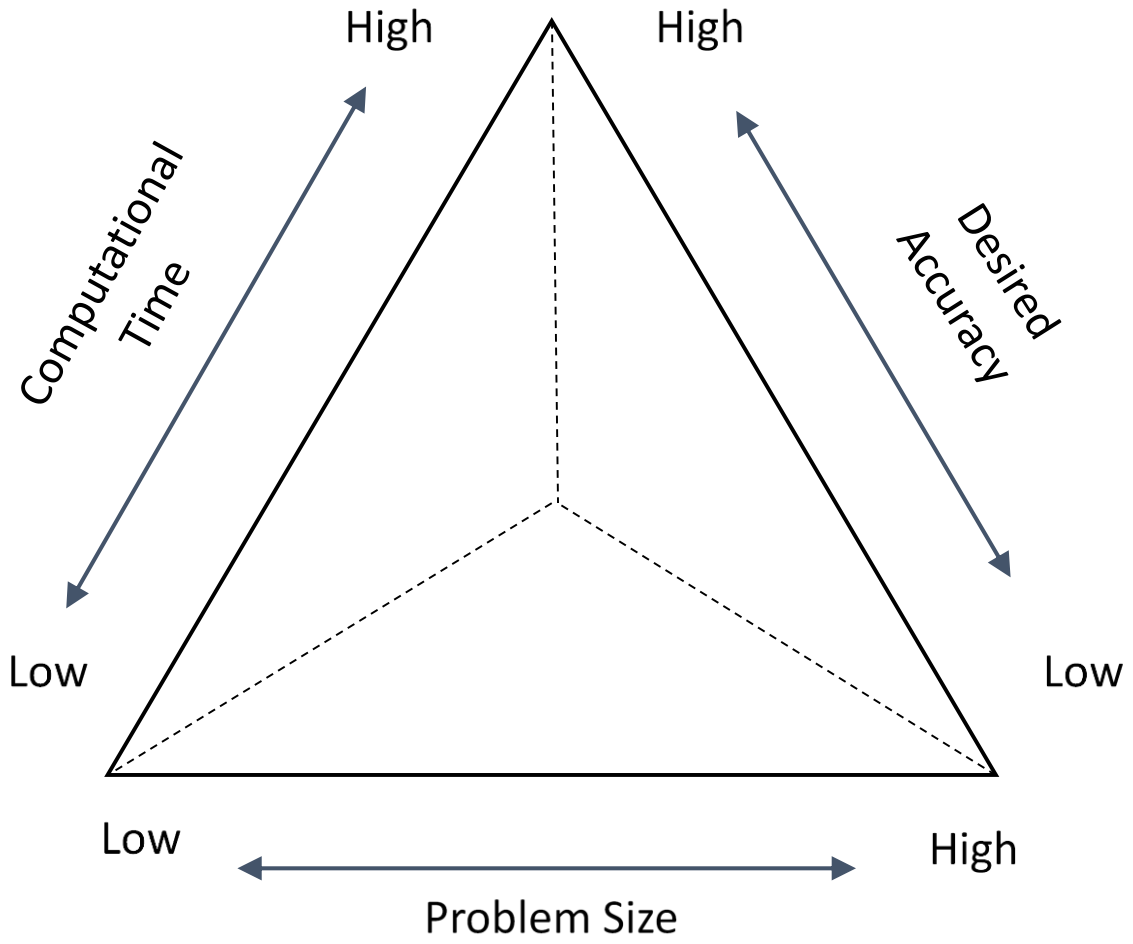


Figure 6.1. Triangle showing the relationships among three design criteria (Alizadeh et al., 2020a).

6.2 Predicting the Microstructure of the Final Rod in a Hot Rod Rolling Problem

To test our hypothesis that an ensemble of surrogates is more accurate than those determined using individual RSM, Kriging, RBF surrogates, we choose a hot rod rolling problem as an example. In this example problem, our interest is to accurately predict the microstructure of the final rod product using surrogate models such that the formation of banded microstructure (microstructure with alternate layers of ferrite and pearlite occurring due to the presence of microsegregates, thereby affecting the mechanical properties) can be managed. The accuracy of the surrogate modeling process is hence important to properly

estimate the final microstructure produced. As shown in Figure 6.2, in the hot rod rolling process problem that we are addressing, a billet of square cross-section having an initial austenitic microstructure is rolled using rollers and further cooled at run-out table to change the shape, microstructure and mechanical properties of the rod.

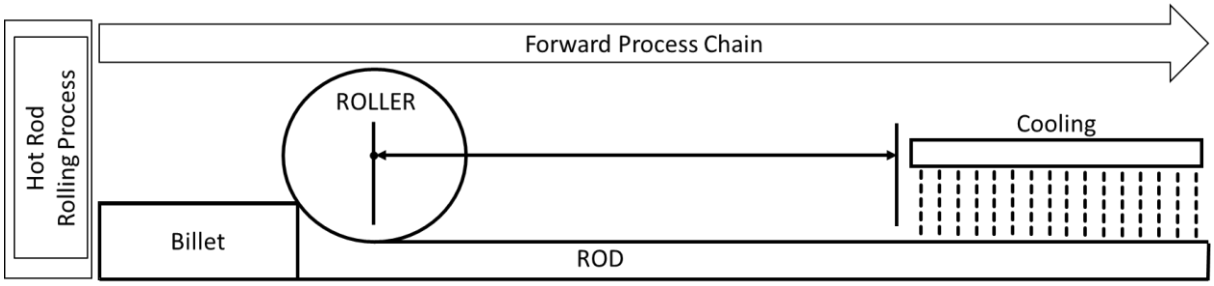


Figure 6.2. Hot rod rolling process (Alizadeh et al., 2019).

In this problem, we are only considering the phase transformation of austenite to the two phases, ferrite and pearlite. We assume in this problem that the phase transformation of austenite to ferrite and pearlite occurs only during the cooling stage after hot rolling. Two types of ferrite phases, namely, Allotriomorphic ferrite and Widmanstatten ferrite are considered in this problem. A slow cooling rate favors the formation of banded ferrite/pearlite microstructure as there is enough time for carbon diffusion and ferrite (allotriomorphic ferrite mostly) nucleation. Suppressing banding is possible via fasting cooling rate, but the elimination of microsegregates, the source for banding, is not possible. In addition to the cooling rate, the initial austenite grain size, percentages of carbon and manganese are important variables in the phase transformation process. For example, a small austenite grain size facilitates the phase transformation to ferrite. A small austenite grain size supports the increase of grain boundary area per volume available for nucleation resulting in more allotriomorphic ferrite nuclei and smaller ferrite grain sizes. This occurs at regions of low manganese concentrations. The rest of austenite transforms to pearlite phase. A high austenite grain size; however, results in an increase in Widmanstatten ferrite. A banded microstructure of ferrite and pearlite formed due to the presence of microsegregates is shown in Figure 6.3.

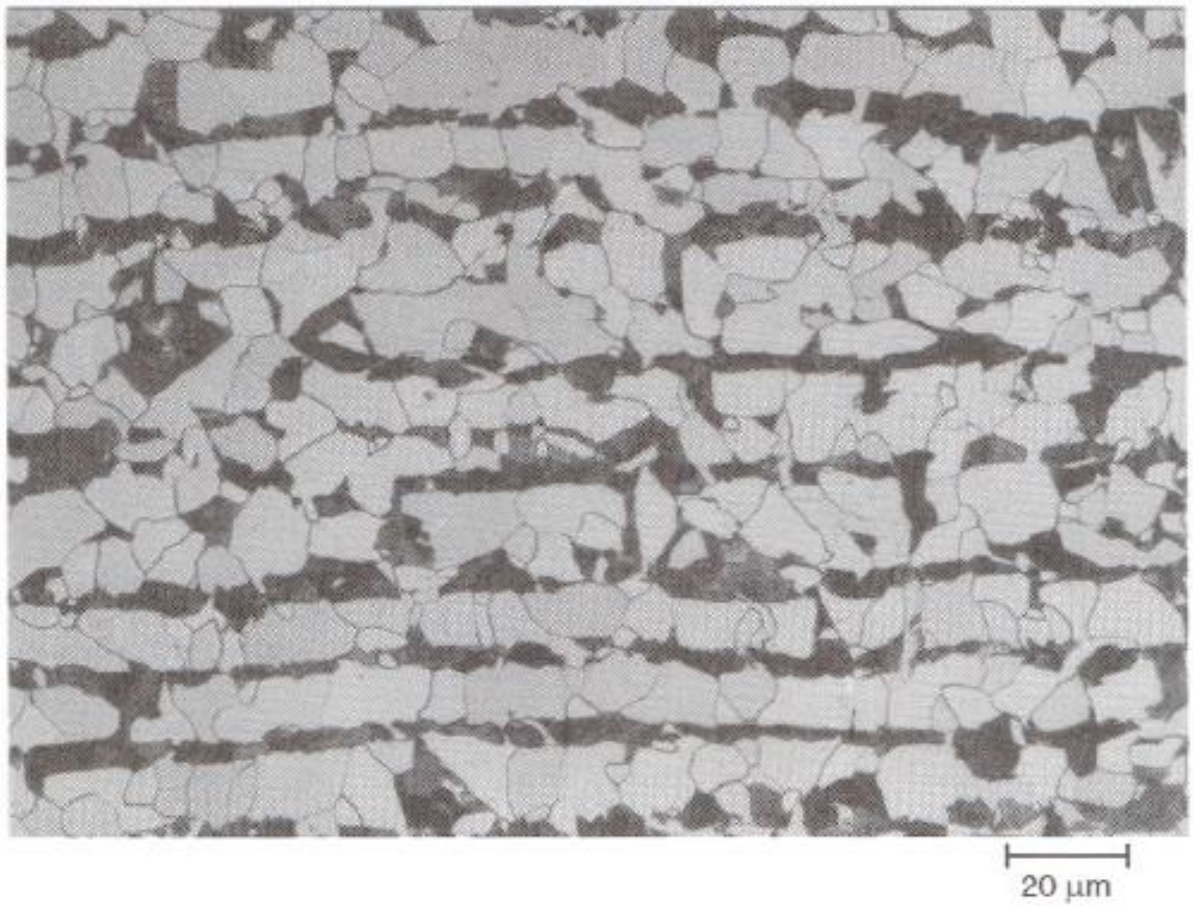


Figure 6.3. An example of a banded microstructure in 1020 steel consisting of ferrite (light) and pearlite (dark) (Jägle, 2007).

In prior studies, authors report the effects of banded microstructures on the mechanical properties of final products, see (Korda et al., 2006; Krauss, 2003; Spitzig, 1983; Tomita, 1995). Hence, it is critical to properly predict the microstructure after phase transformation to manage the banded microstructure such that the mechanical properties of the product can be controlled. The ferrite-pearlite banded microstructure is primarily caused due to microsegregates in the form of alloying elements of manganese, Sulphur, etc., that are embedded into the steel during the solidification process after casting. The initial austenite grain size, cooling rate, carbon concentration, and manganese concentration are selected in this study based on literature review as the major factors influencing the final microstructure phase formed after rolling and cooling. The program STRUCTURE developed by Jones and Bhadeshia is used as the simulation program to predict the microstructure phases (Jones and

Bhadeshia, 2017). The other input fixed parameters for the simulation program like the austenite-ferrite interfacial energy, activation energy for atomic transfer, aspect ratio for nucleation of ferrite, fraction of effective boundary sites, total volume fractions of inclusions, nucleation factor for pearlite and aspect ratios of growing allotriomorphic ferrite, Widmanstatten ferrite and pearlite are selected and defined based on the values reported by (Jägle, 2007). The control factors thus considered in this work are described in Table 6.3. The output of the simulation includes volume fractions of pearlite, and two types of ferrites, namely, allotriomorphic ferrite and Widmanstätten ferrite for the different values of each of the four input variables. Therefore, the simulation addressed in this problem involves four input variables and three output variables.

A fractional factorial design of experiments to generate response data sets is carried out by Nellippallil and co-authors (Nellippallil et al., 2018) using the simulation program, STRUCTURE developed by Jones and Bhadeshia (Jones and Bhadeshia, 1997). Polynomial response surface models are fit for each of the responses.

Table 6.3. The design variables

Design Variable	**Definition**
Manganese concentration after rolling (Mn)	(Jones and Bhadeshia, 1997) point out that manganese is an austenite stabilizing agent. Therefore, transformation to ferrite occurs at low manganese regions. Due to this, the high manganese region gets enriched with carbon leading to the formation of pearlite.
Final Austenite grain size after rolling (AGS)	This parameter has an inbuilt effect on grain boundary area per unit volume and thus on nucleation itself. Because of this effect, and the simultaneous phase transformations, the average grain size (neglecting the length scale) have a major bearing on final microstructure.
Cooling rate	Banding is usually suppressed by high cooling rates. Lower cooling rates favor carbon diffusion leading to the development of banded microstructure.
Carbon content	The carbon content changes the physical properties of commercially available steel and hence determine which component is formed first during the initial stages of cooling

Nellippallil and co-authors explain how different values of the four input variables, cooling rate, carbon concentration, manganese concentration, and austenite grain size, affect the final microstructural phases (pearlite, allotriomorphic ferrite, and Widmanstätten ferrite) based on the polynomial response surface model developed (Nellippallil et al., 2018). Nellippallil and co-authors verify the model predictions by comparing them with experimentally measured data reported by Bodnar and Hensen *(Bodnar and Hansen, 1994)*. We use the fractional factorial design of experiments data set by Nellippallil and co-authors in this work. In Figure 6.4, we show the comparison of polynomial response surface model predictions by Nellippallil and co-authors (Nellippallil et al., 2018) with the phase fractions reported in the literature by Bodnar and Hensen *(Bodnar and Hansen, 1994)*. We observe from Figure 6.4 that the model predictions lie in the vicinity of the straight line depicting the measured values.

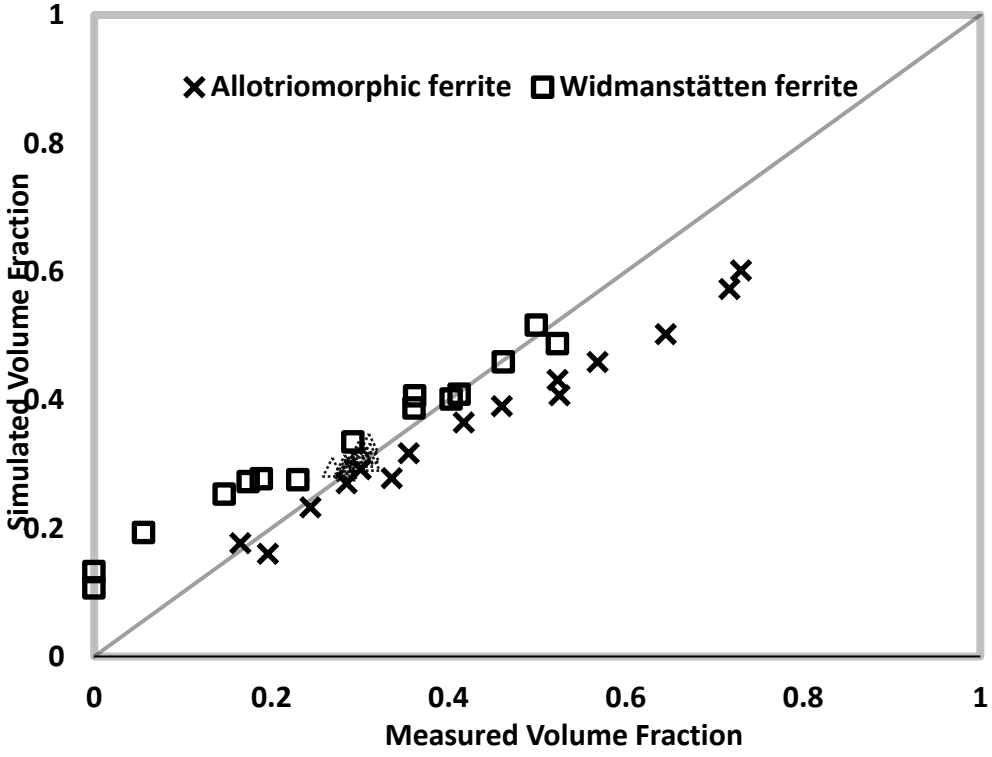


Figure 6.4. Comparison of polynomial response surface model predictions *(Nellippallil et al., 2018).*

6.3 Method for Selecting the Surrogate Model Based on Time, Size, and Accuracy

As shown in , the procedure consists of several steps. The first step is to find some criteria to choose the most appropriate surrogate model. Based on a comprehensive critical literature review, a balanced triangle of three important characteristics of the experiment (accuracy, size and time) are presented in Figure 6.1 (Alizadeh et al., 2020a). The second step is to build surrogate models, including RSM, Kriging, RBF and an ensemble of them. In this step, we compare the performance of these models for the Hot Rod Rolling problem. In the third step, the outcome of the second step is summarized in a sort of database to be used in the future.

6.3.1. Surrogate modeling process

The method for selecting the surrogate model based on time, size and accuracy are illustrated in . After identifying the problem in Section 6.2, the next step is to generate

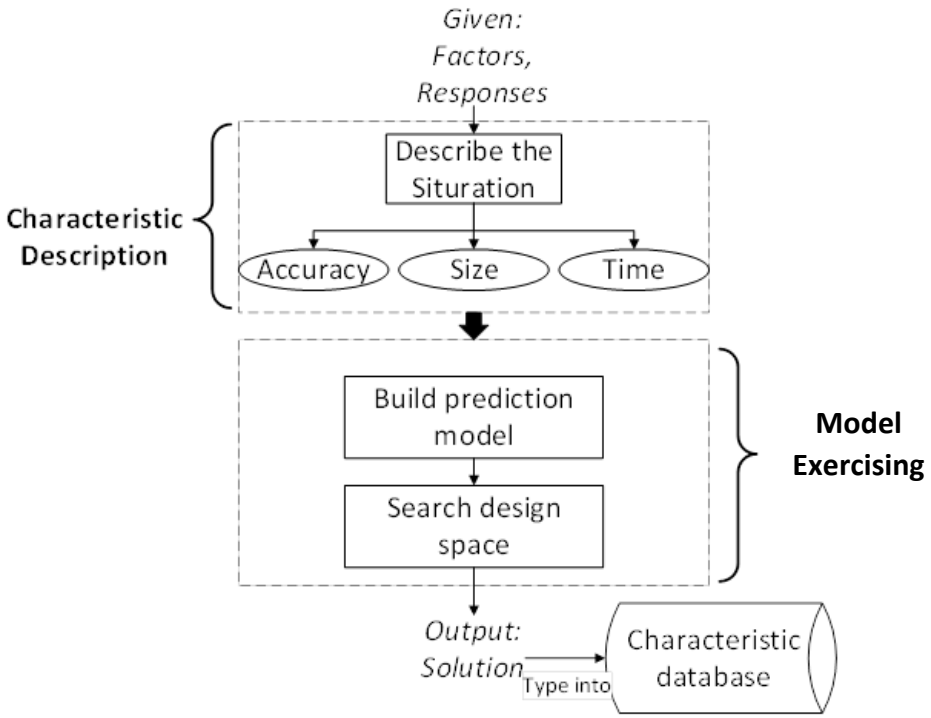


Figure 6.5. Method for selecting the surrogate model based on time, size, and accuracy.

sampling data. The design of experiments (DOE) is used to obtain the sampling data over the desired range of input variables. In the second step, cross-validation (CV) is used to divide the design data into 'training data' and 'testing data'. The 'training data' is used to develop different surrogate models, and the 'testing data' is used as unknown data to estimate the performance of different surrogate models.

Next, different surrogate modeling methods are used to fit the training design data generated and develop a predictive model (in Section 6.3). Finally, the surrogates are evaluated using the root mean square error (RMSE) of the response deviation between the prediction data and the test data (in Section 6.4).

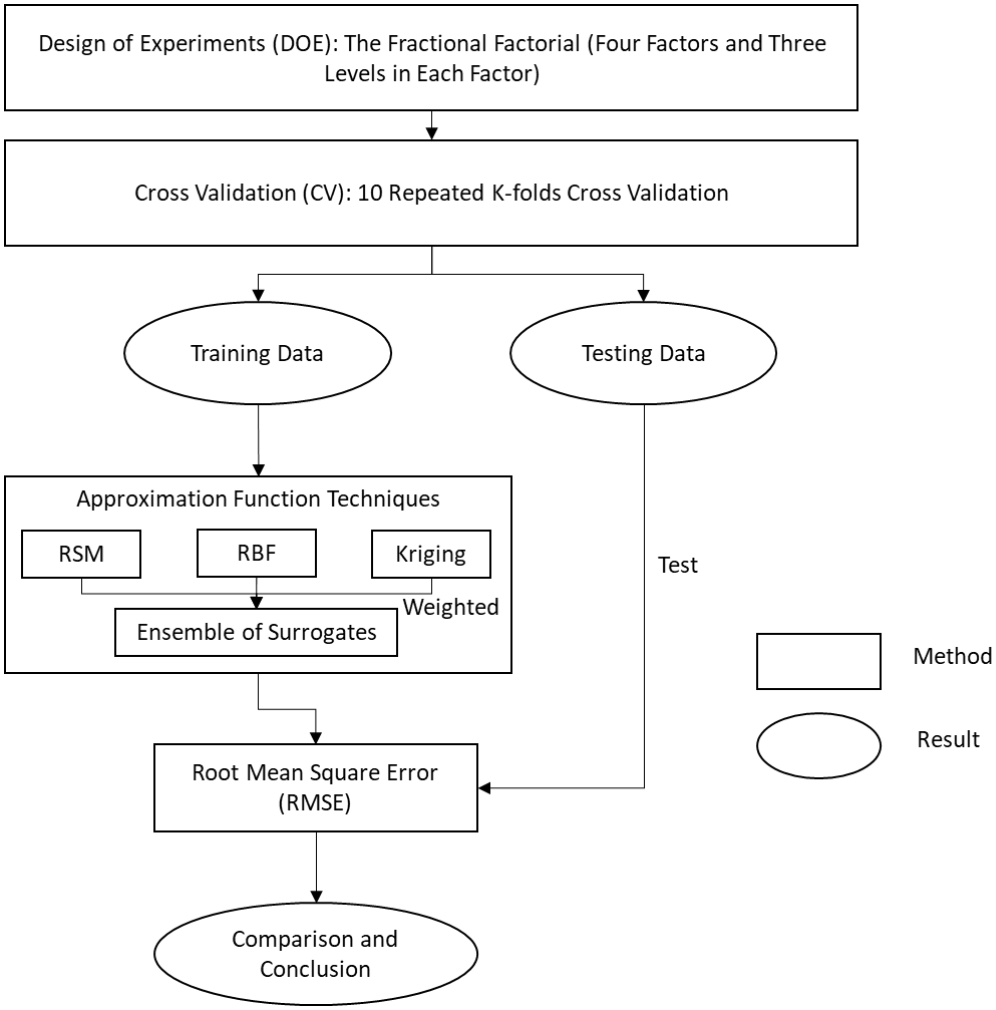


Figure 6.6. The experimental procedure.

6.3.2. Design of experiments and cross-validation

In this work, we use the data generated by (Nellippallil et al., 2018) using a fractional factorial design (Montgomery, 2017) to organize the experiments. Considering the trade-off between the cost of evaluation of the original simulation and the increase of fidelity associated with the increasing number of sampling data, three levels including the upper, lower and middle points of the range in each factor are selected to manage the sampling process. The factors and factor levels are shown in Table 6.4.

Table 6.4. Factors and factor levels for DOE

Level	***CR*** *K/min*	**AGS** μm	***[C]*** %	***[Mn]*** %
1	11	30	0.18	0.7
2	55	55	0.24	1.1
3	100	100	0.3	1.5

Now, in order to estimate the performance of the predictive model, K-fold cross-validation is used. $K-1$ equally sized randomly selected subsamples are used as training data and the remaining single subsample is retained to test the model. The training process for each fold is repeated K-1 times, with each subsample being used exactly once as test data. Thus, all observations are fully used for both training and testing. We repeat the whole cross validation process ten times, and the result is the average value of 10 runs of cross-validation. In this way, the impact of noise can be minimized. The next step is to use the training data to construct surrogates, in this case, RSM, KRG, RBF, and an ensemble of surrogates and use the testing data for evaluation.

6.3.3. Function fitting using different surrogate models

Several function approximation techniques are used as surrogates. (Shyy et al., 2001) used RSM and RBF to rocket engine design and compare the prediction of alternative models. It turns out surrogate models have good performance in prediction work. (Zerpa et al., 2005)

integrate RSM, KRG, and RBF as an ensemble of the surrogate model and apply it into an alkaline-surfactant-polymer flooding processes. (Bellucci and Bauer Jr, 2017) use RSM, KRG, and RBF to make robust parameter design. So, in this work, we choose RSM, KRG, RBF, which are commonly used in the previous chapters, and their combination as an ensemble of surrogate models to develop different prediction models.

6.3.4. Fitting a response surface model to the dataset

The response surface method (RSM) is also known as the polynomial regression method which has the simplest of parameters (that is, coefficients in a polynomial function) and is calculated using least squares regression (Razavi et al., 2012). In this work, we use a second-order polynomial function as the surrogate model.

If $\boldsymbol{x}$ is an independent vector of factors, $\boldsymbol{y}$ is the vector of responses, the impact of $\boldsymbol{x}$ on $\boldsymbol{y}$ and their relationship can be illustrated as follows:

$$\hat{\boldsymbol{y}}(\boldsymbol{x}) = \alpha_0 + \sum_{i=1}^{n} \alpha_i x_i + \sum_{i=1}^{n} \alpha_{ii} x_i^2 + \sum \sum_{i<j} \alpha_{ii} x_i \, x_j \qquad \text{Equation 6.1}$$

where α represents the coefficients of the polynomial function and n is the number of independent factors.

6.3.5. Fitting a Kriging model to the dataset

Kriging is a type of interpolating technique which is a polynomial model of an input vector of factors $\boldsymbol{x}$, $f(\boldsymbol{x})$, and localized deviation of $\boldsymbol{x}$, $g(\boldsymbol{x})$, as follows:

$$\hat{y}(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{x}) \qquad \text{Equation 6.2}$$

$f(\boldsymbol{x})$ is the polynomial term, which is a global function over the entire input space (Razavi et al., 2012). Usually, $f(\boldsymbol{x})$ is a constant number or a linear polynomial function. And $g(\boldsymbol{x})$ is

a localized deviation function or can be called the basic function. In this article, the focus is on one type of Kriging called Ordinary Kriging (Kleijnen, 2017),

$$\hat{y}(\boldsymbol{x}) = \mu + g(\boldsymbol{x}) \qquad \text{Equation 6.3}$$

where μ is an unknown constant, which represents the simulation output averaged over the experimental area and $g(\boldsymbol{x})$ is a zero-mean stochastic process.

6.3.6. Fitting a radial basis function model to the dataset

The radial basis function (RBF) technique is based on a mathematical function and its value is calculated based on the distance the between origin and each point (Montgomery, 2017).

$$r_{i,j} = r(\boldsymbol{x_i}, \boldsymbol{x_j}) = \|\boldsymbol{x_i} - \boldsymbol{x_j}\| \qquad \text{Equation 6.4}$$

Where $r_{i,j}$ denotes the Euclidean distance between two different sample points. Radial functions are employed to connect the distance r with the outputs, then the integration of these functions is used to estimate complicated mathematical functions. These functions are then used in constructing the surrogate models:

$$\hat{y}(\boldsymbol{x_{new}}) = \sum_{i=1}^{M} r_i Q(\|\boldsymbol{x_{new}} - \boldsymbol{x_i}\|) \qquad \text{Equation 6.5}$$

The surrogate function $\hat{y}(\boldsymbol{x_{new}})$ are an integration of **M** radial basis functions and each of them is linked to a distinct x_i and has a weight of r_i (Mirjalili, 2019). In this work, the Gaussian function is selected as the radial basis function.

6.3.7. Creating an ensemble of surrogate models using different surrogate models

We use the weighted average model proposed by (Goel et al., 2007) to create the ensemble of surrogates. As we just use three individual surrogate models, the predicted response of the ensemble of surrogate model is:

$$\hat{y}_{EoS} = \sum_{i}^{N_{SM}} w_i \hat{y}_i = w_{RSM}\hat{y}_{RSM} + w_{KRG}\hat{y}_{KRG} + w_{RBF}\hat{y}_{RBF} \quad \text{Equation 6.6}$$

where w_i is the weight of each individual surrogate and $\hat{y}_i$ is the predicted response of the i^{th} individual surrogate. For the selection of weights, a strategy which is based on generalized mean square cross-validation error (GMS error) is proposed in (Goel et al., 2007).

$$w_i{}^* = \left(E_i + \alpha E_{avg}\right)^{\beta},\ w_i = w_i{}^* / \sum_i w_i{}^*\, E_{avg} = \sum_i^{N_{SM}} E_i / N_{SM} \quad \text{Equation 6.7}$$

Where E_i denotes the GMS error, N_{SM} is the number of sample points, and two constraints, α, β are chosen to be $\alpha = 0.05$ and $\beta = -1$ respectively (Samad et al., 2006).

6.3.8. Prediction metric: root mean square error

In evaluating the process, the actual objective function value for the testing data is known and regarded as the target value. This value is then used to calculate the error at all testing points. The root means square error (RMSE) is used as the prediction metric to assess the accuracy of each prediction model generated by different surrogates. The definition of RMSE is:

$$RMSE = \sqrt{\sum_{i=1}^{N_{test}} (y_i - \hat{y}_i)^2 \Big/ N_{test}} \qquad \text{Equation 6.8}$$

Where N_{test} represents the number of testing points, y_i and $\hat{y}_i$ are the actual response and the predicted value of the response from a surrogate. The lower the RMSE, the greater the accuracy of the surrogate model, and vice versa.

6.4 Results and Discussion on Hot Rod Rolling Problem

6.4.1. Comparison between ensembles of surrogates and individual surrogates

Appendix D contains the sample points for the three objectives, allotriomorphic ferrite (Y1), Widmanstätten ferrite (Y2) and pearlite (Y3). In order to generate test data to validate the performance of these surrogate models, we repeat a 9-fold cross-validation process 10 times. In each run, all data sets are randomly partitioned into 9 subsamples (groups). Of the 9 subsamples, one subsample is used as the testing data set and the remaining eight subsamples are used to train the model. Through 9 repetitions, all observations are involved in training and testing. Three outputs are allotriomorphic ferrite (Y1), Widmanstätten ferrite (Y2) and pearlite (Y3) of steel. In two upper plots of Figure 6.7, we indicate the RMSE for the prediction of output Y1 (allotriomorphic ferrite) and output Y2 (Widmanstätten ferrite) respectively. In the two lower plots of Figure 6.7, we represent the RMSE of output Y3 (pearlite). The difference is that in the left one we show the errors of four surrogates, in the right one we show the expanded version for the ensemble of surrogates, Kriging, and response surface models. The experimental results for three outputs from the hot rod rolling problem with response surface method (RSM), Kriging (KRG) and radial basis function (RBF) and an ensemble of surrogates (EoS) are shown in Figure 6.7.

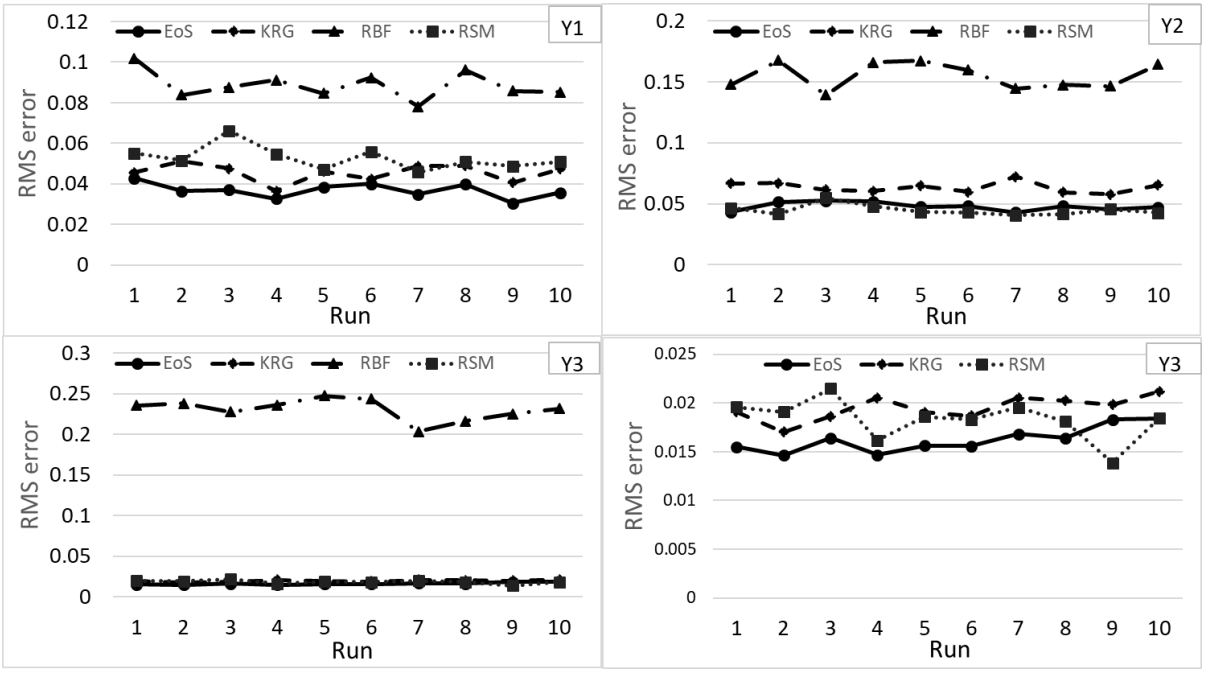


Figure 6.7. RMSE in the prediction of surrogates in 10 runs.

The result shows that RBF gives the highest RMS error for all three objectives, especially for output Y3 (Pearlite), the error of the RBF is almost five times larger than other three surrogates, which means RBF has the lowest accuracy in predicting all three objectives. Also, the difference between the ensemble of surrogates, Kriging, and RSM is minimal, but the ensemble of surrogates shows subtle advantages in accuracy. As for the quantitative validation of the surrogates, a statistical analysis of RMS errors of surrogate predictions for ten runs in the design space is shown in

Table 6.5. EoS has the best accuracy except for Y2 (Widmanstätten ferrite), and RBF is the least accurate. For Y2, EoS has the second-highest accuracy, which is second to RSM and the difference between these two is very small ($(0.0478 - 0.0447)/0.0478 \approx 6.5\%$).

This indicates that an ensemble of surrogates has the most appropriate performance for the hot rod rolling problem because of its accuracy and relatively insensitive predictions to the number of data points for all three objectives and therefore ensembles of surrogates are more robust.

Table 6.5. Statistical analysis of RMSE value by various surrogates in 10 runs.

Surrogate	RMS errors produced by the surrogates		
	F1 Allotriomorphic ferrite	F2 Widmanstätten ferrite	F3 Pearlite
EoS	**3.68E-02**	4.78E-02	**1.62E-02**
KRG	4.55E-02	6.35E-02	1.95E-02
RBF	8.86E-02	1.55E-01	2.31E-01
RSM	5.27E-02	**4.47E-02**	1.83E-02

6.4.2. Trade-offs among accuracy, size and time

To find a relationship among the size, accuracy, and computation time of surrogates for the hot rod rolling problem, we change the size of the problem in terms of the number of training data which are used to train the prediction model. As the sample data has 27 points and the K-fold cross-validation (CV) is used to generate the training data, we increase training data by the way of decreasing the fold value. The relationship between the fold number and the test data can be expressed as a function:

$$N_{test} = 27 - {}^{27}/_{N_{fold}} \qquad \text{Equation 6.9}$$

Where N_{test} is the number of test data, N_{fold} is the fold number. Thus, when the fold value decreases from 9 folds to 2 folds, the training data is reduced from 24 to 14. In order to make a fair comparison between four different surrogates, the model training process is organized with the same training data. The comparison between these four surrogates in three outputs of the hot rod rolling problem is shown in Figure 6.8.

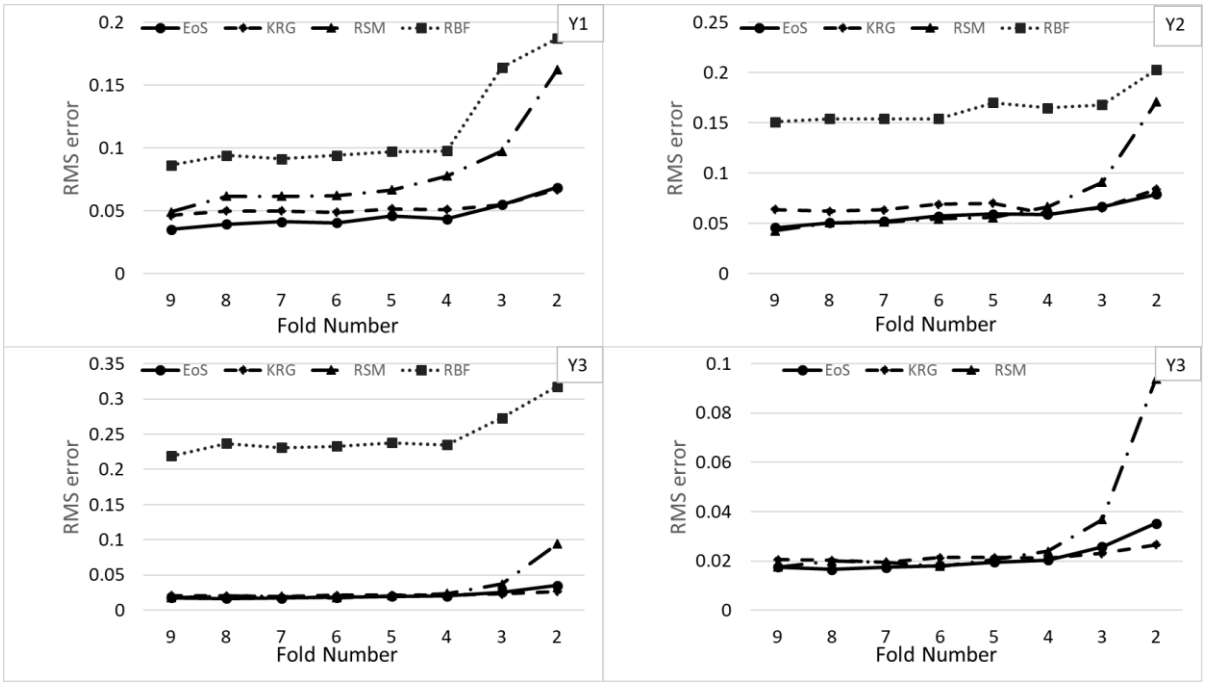


Figure 6.8. RMS errors in the prediction of surrogates with different folds of data from 9 to 2.

Note: with the upper two plots in Figure 6.8, we indicate the RMS errors of predicting Y1 (allotriomorphic ferrite) and output Y2 (Widmanstätten ferrite). With the lower two plots of Figure 6.8, we represent the RMS errors of output Y3 (pearlite). The difference is that with the left one we show the errors of four surrogate and with the right one we only show three (expect RBF).

From Figure 6.4, for each surrogate model, the RMS error gradually increases as the amount of training data decreases, therefore accuracy has a negative correlation with the number of samples used in the training data, and this trend is in line with our intuition. In addition, it is interesting that the error curves of each surrogate all have a smooth change from the 9-fold to 4-fold training data but abruptly increase at the 3- and 2-fold data, especially the RSM and RBF. That is because, with small numbers of training data (2 and 3-fold), surrogate models are more prone to have low fidelity compared to the associated physical problem. As for the comparison between these four surrogates, EoS, KRG and RMS have relatively lower RMS error and higher accuracy than RBF for all folds. EoS and KRG generally show very robust behavior, and RSM also shows a robust behavior until 2 and 3-fold data are used, where there is a sharp surge in error and the RSM's accuracy goes down immediately.

Table 6.6. Program run time to compute values for four surrogates from 9 to2-fold training data for output Y1.

No. of folds	Program run time for different surrogate models (s)			
	EoS	KRG	RSM	RBF
9	115.5	114.2	0.21	**0.098**
8	102.3	101.3	0.158	**0.072**
7	86.21	84.12	0.132	**0.06**
6	67.01	65.76	0.116	**0.059**
5	46.69	45.29	0.092	**0.051**
4	44.21	40.31	0.082	**0.039**
3	36.49	35.21	0.072	**0.031**
2	23.69	21.56	0.027	**0.022**

The program run time is the computational time required by Python codes in a Lenovo computer i7-4720HQ 8G 128G SSD+1T GTX960M.

As can be seen in Figure 6.6, RBF has the lowest program run time among the four surrogates. However, from a practical perspective, the time should include not only the program run time which is consumed to construct the surrogate model but also the simulation time used for sampling. So, in this work, the time is defined as follows.

$$T = T_{pm} + T_{sm} \qquad \text{Equation 6.10}$$

Where, T_{pm} is the program run time and T_{sm} is the simulation time.

In the hot rod rolling problem, the average sampling time for generating each data point in the simulation is about 3.5 hours (12600s), which is at least 2 orders of magnitude larger than

the program run time (the longest one is about 150s) for all four surrogates. So, the program run time can be ignored in Equation 6.10, and the time we mention in the following discussion is the equivalent of the sampling time, as shown in the following.

$$T \approx T_{sample} = 3.5h \times N_{sample} \qquad \text{Equation 6.11}$$

Where the N_{sample} is the number of sample points. Therefore, the time is proportional to the number of sample points. In order to generate selection rules, a detailed comparison of results among the different surrogate models for each output is shown in the following three sections.

6.4.3. Results for output Y1 (allotriomorphic ferrite)

The RMS errors, namely, the accuracy, for each surrogate for different numbers of design data from 9 folds to 2 folds are shown in Figure 6.8. Ensembles of surrogates have relatively lower RMS errors and therefore higher accuracy than three individual surrogates for the 9, 8, 7, 6, 5- and 4-fold data. In addition, ensembles of surrogates in 2 and 3-fold data have almost equal accuracy (The difference is about 0.001.) and 2 and 3-fold Kriging have much higher accuracy than RSM and RBF. Both Kriging and an ensemble of surrogates have higher accuracy than RBF in all k-fold cross-validation methods. Also, they have higher accuracy even when we use 2-fold cross validation for them and 9-fold cross validation for RBF. So, RBF is not a good choice regarding the accuracy and size measures.

In Table 6.7, we the lowest values of error for each number of folds in bold. We show that the RMS error gradually decreases with decreasing training data for each surrogate, which means that the accuracy is negatively correlated with the amount of training data, and this trend is in-line with our intuition.

Table 6.7. RMS errors generated by four different surrogates from 9-2 fold training data for output Y1.

	RMS errors for different surrogate models			
No. of folds	EoS	KRG	RSM	RBF
9	**3.52E-02**	4.65E-02	4.91E-02	8.63E-02
8	**3.96E-02**	4.99E-02	6.16E-02	9.41E-02
7	**4.14E-02**	5.01E-02	6.17E-02	9.15E-02
6	**4.05E-02**	4.88E-02	6.22E-02	9.41E-02
5	**4.62E-02**	5.18E-02	6.66E-02	9.71E-02
4	**4.36E-02**	5.12E-02	7.77E-02	9.78E-02
3	5.50E-02	**5.46E-02**	9.74E-02	1.64E-01
2	6.86E-02	**6.70E-02**	1.62E-01	1.87E-01

In addition, an ensemble of surrogates has much greater accuracy than the other three individual surrogates trained for 9-fold data even when we use 8, 7, 6, 5, and 4-folds instead of 9-fold data. So, an ensemble of surrogates gives the most accurate predictions and needs the lowest time to obtain the prediction for Allotriomorphic Ferrite when we use -9 to 4-fold training data. Also, RBF is not a good choice because time, accuracy, and size measures for 9-fold cross validation for RBF is not greater than the 2-fold cross validation for the ensembles of surrogates and Kriging. As for the comparison between RSM and EoS, EoS has greater accuracy than RSM in -4 to 9-fold cross-validation. When we use 3-fold data to train the ensemble of surrogates, we obtain higher accuracy than when we use 9-fold of data to train RSM. Also, we obtain higher accuracy by using -5 to 8-fold of RSM rather than using 2-fold of data to train the EoS, while RSMs trained with 2 to 4-fold of data have lower accuracy than EoS trained with 2-fold data. Based on this analysis, some recommendations for time and size of the dataset along with the accuracy of Output 1 (Y1) can be developed. If we consider the

ensemble of surrogates individually, by using 2-fold instead of 9-fold, accuracy increases about $\frac{0.07-0.035}{0.07} = 50\%$. On the other hand, by using 2-fold instead of 9-fold, we need ((27-27/2=13) *3.5=45.5) hours instead of 27*3.5=94.5 hours, and so, there are a savings of 35 hours. As for the comparison between ensembles of surrogates and individual surrogates, ensembles of surrogates are the recommended because of their higher accuracy and low time-consumption for 4 to 9 folds (19-24) training data, whereas RBF is always the worst choice.

6.4.4. Results for output Y2 (Widmanstätten ferrite)

For Widmanstätten ferrite, the ensemble of surrogates has relatively lower RMS errors, and so, higher accuracy than Kriging, and RBF even when we use 9, 8, 7, 6, 5, 4, 3 and 2 folds of the data. Although RSM has a relatively higher accuracy for 9, 8, 7, 6, and 5 folds of data, again it has lower accuracy in 2 and 3 folds in comparison to the ensemble of surrogates. RBF also again has relatively low accuracy in comparison to other surrogate modeling methods. Kriging also has relatively less accuracy for 5 to 9 folds. RSM and has only greater accuracy than RSM in 2 to 4 folds. Interestingly, RSM with fewer data points, including 5, 6, 7, and 8 folds has better accuracy than 9-fold Kriging. From the analysis in Table 6.10, some recommendations considering time and size of the dataset along with the accuracy for Widmanstätten Ferrite (Y2) are developed. Entries in bold show the lowest values of error for each number of folds. If we consider the ensemble of surrogates by itself, by using 2-fold instead of 9-fold, accuracy increases about $\frac{0.08-0.035}{0.08} = 0.43\%$. On the other hand, by using 2-fold instead of 9-fold data, we need 45.5 hours instead of 94.5 hours, and so, we can save 35 hours. Also, to compare ensembles of surrogates with other single surrogates, by using a 4-fold EoS we need 70.87 hours instead of 94.5 hours when we choose to use 9-fold RSM, and so, we can save 23.63 hours (23.63 hours faster surrogate) and have $\frac{0.055-0.04}{0.055} = 27\%$ less accuracy.

Table 6.8. RMS errors generated by four different surrogates from 9-2 fold train data for output Y2.

	RMS errors of different surrogate models			
No. of folds	EoS	KRG	RSM	RBF
9	4.59E-02	6.38E-02	**4.26E-02**	1.51E-01
8	5.07E-02	6.23E-02	**5.07E-02**	1.54E-01
7	5.22E-02	6.34E-02	**5.13E-02**	1.54E-01
6	5.72E-02	6.90E-02	**5.45E-02**	1.54E-01
5	5.95E-02	7.03E-02	**5.60E-02**	1.70E-01
4	**5.91E-02**	5.99E-02	6.66E-02	1.65E-01
3	**6.66E-02**	6.67E-02	9.11E-02	1.68E-01
2	**7.92E-02**	8.42E-02	1.71E-01	2.03E-01

6.4.1. Results for output Y3 (pearlite)

As shown in the summary of the results is given in Table 6.8. The highlighted values denote to the lowest RMS error obtained from the corresponding surrogate model. As illustrated in Table 6.10, for the first response variable, Y1 (Allotriomorphic Ferrite), in 4-9 folds of data, an ensemble of surrogates (EoS) has lower RMSE than individual surrogates and only for 2 and 3 folds of data, Kriging has lower RMSE than others. Also, for the second response variable, Y2 (Widmanstätten Ferrite), in 2-4 folds of data, EoS has lower RMSE than other surrogates. Finally, for the third response variable, Y3 (Pearlite), in 4-9 folds 9, there is a slight difference between the accuracy of EoS, KRG, and RSM except for 2 and 3-fold RSM which has much lower accuracy. On the other hand, there is a huge gap among these three surrogates and RBF, which has very low accuracy and can hardly be compared with others. If we consider surrogates individually, there is almost no difference among using 4, 5, 6, 8, 9 folds of KRG and EoS. Therefore, it is possible to save 23.63 hours by using 4-fold Kriging or and an

ensemble of surrogates instead of 9-fold Kriging or and an ensemble of surrogates. Also, for RSM, there is a very slight difference between using 4-fold instead of 9-fold. While using 3-fold RSM, the accuracy reduces $\frac{0.035-0.03}{0.035} = 14\%$, but the process is much faster. In other words, by using 3-fold instead of 9-fold, 63 hours instead of 94.5 hours are needed, which can save 31.5 hours. It also means that we can use only 9 data points instead of 27 data points which saves 67% of the simulation time which we need to spend to generate the simulation data.

Table 6.9. RMS errors generated by four different surrogates from 9-2 fold train data for output Y3.

No. of folds	RMS errors of different surrogate models	RMS errors of different surrogate models	RMS errors of different surrogate models	RMS errors of different surrogate models
	KRG	KRG	KRG	KRG
9	EoS	KRG	RSM	RBF
8	**1.75E-02**	2.05E-02	1.75E-02	2.19E-01
7	**1.66E-02**	2.02E-02	2.00E-02	2.37E-01
6	**1.74E-02**	1.94E-02	1.95E-02	2.31E-01
5	1.81E-02	2.14E-02	**1.78E-02**	2.33E-01
4	**1.95E-02**	2.13E-02	2.06E-02	2.38E-01
3	**2.04E-02**	2.12E-02	2.39E-02	2.35E-01
2	2.57E-02	**2.31E-02**	3.68E-02	2.73E-01

The summary of the results is given in Table 10. The highlighted values denote to the lowest RMS error obtained from the corresponding surrogate model. As illustrated in Table 6.10, for the first response variable, Y1 (Allotriomorphic Ferrite), in 4-9 folds of data, an ensemble of

surrogates (EoS) has lower RMSE than individual surrogates and only for 2 and 3 folds of data, Kriging has lower RMSE than others. Also, for the second response variable, Y2 (Widmanstätten Ferrite), in 2-4 folds of data, EoS has lower RMSE than other surrogates. Finally, for the third response variable, Y3 (Pearlite), in 4-9 folds of data, EoS has lower RMSE than other surrogates. the third response variable, Y3 (Pearlite), in 4-9 folds of data, EoS has lower RMSE than other surrogates.

Table 6.10. RMS errors generated by four surrogates from 9-2 fold training data for output Y1, Y2, and Y3.

	RMS errors for different surrogate models											
	Y1 (Allotriomorphic Ferrite)				Y2 (Widmanstätten Ferrite)				Y3 (Pearlite)			
No. of folds	EoS	KRG	RSM	RBF	EoS	KRG	RSM	RBF	EoS	KRG	RSM	RBF
9	**3.52E-02**	4.65E-02	4.91E-02	8.63 E-02	4.59 E-02	6.38 E-02	**4.26 E-02**	1.51 E-01	**1.75 E-02**	2.05 E-02	1.75 E-02	2.19 E-01
8	**3.96E-02**	4.99E-02	6.16E-02	9.41 E-02	5.07 E-02	6.23 E-02	**5.07 E-02**	1.54 E-01	**1.66 E-02**	2.02 E-02	2.00 E-02	2.37 E-01
7	**4.14E-02**	5.01E-02	6.17E-02	9.15 E-02	5.22 E-02	6.34 E-02	**5.13 E-02**	1.54 E-01	**1.74 E-02**	1.94 E-02	1.95 E-02	2.31 E-01
6	**4.05E-02**	4.88E-02	6.22E-02	9.41 E-02	5.72 E-02	6.90 E-02	**5.45 E-02**	1.54 E-01	1.81 E-02	2.14 E-02	**1.78 E-02**	2.33 E-01
5	**4.62E-02**	5.18E-02	6.66E-02	9.71 E-02	5.95 E-02	7.03 E-02	**5.60 E-02**	1.70 E-01	**1.95 E-02**	2.13 E-02	2.06 E-02	2.38 E-01
4	**4.36E-02**	5.12E-02	7.77E-02	9.78 E-02	**5.91 E-02**	5.99 E-02	6.66 E-02	1.65 E-01	**2.04 E-02**	2.12 E-02	2.39 E-02	2.35 E-01
3	5.50E-02	**5.46E-02**	9.74E-02	1.64 E-01	**6.66 E-02**	6.67 E-02	9.11 E-02	1.68 E-01	2.57 E-02	**2.31 E-02**	3.68 E-02	2.73 E-01

RMS errors for different surrogate models												
	Y1 (Allotriomorphic Ferrite)				Y2 (Widmanstätten Ferrite)				Y3 (Pearlite)			
2	6.86E-02	**6.70E-02**	1.62E-01	1.87 E-01	**7.92 E-02**	8.42 E-02	1.71 E-01	2.03 E-01	3.52 E-02	**2.66 E-02**	9.39 E-02	3.18 E-01

In comparison to some other studies such as Viana and coauthors (2013), Chaudhuri and Haftka (2014), Badhurshah and Samad (2015) and Bhattacharjee and coauthors (2018) in ensemble surrogates related issues there is a big difference in the method. Authors in these chapters applied multiple surrogate methods for multiobjective optimization, while in this chapter, we used an ensemble of surrogates for prediction. Also, Goel and coauthors (2007) and Bishop (1995) creates a weighted average surrogate by estimating the covariance between surrogates from residuals at test or training datasets and using the PRESS weighted average surrogate, while we create the weights based on cross-validation errors. Basudhar (2012), Viana and coauthors (2013), Bhattacharjee and coauthors (2016) and Ezhilsabareesh and coauthors (2018) used an ensemble of surrogates in a relatively high number of data while we use an ensemble of surrogates in small data size.

In addition, the ensembles created by Chaudhuri and Haftka (2014), Wang and coauthors (2016), Bhattacharjee and coauthors (2018), Lv and coauthors (2018), Song and coauthors (2018), and Yin and coauthors (2018) are more time consuming than each individual surrogate they used to create the ensemble. Whereas we create ensembles which are less time consuming than individual surrogates and have less inconsistency. In order to give specific guidance for surrogate selection, a rule-based template is given in Table 6.11.

Table 6.11. Specific guidance for the selection of surrogate models for output Y1 (Allotriomorphic Ferrite)

First Selection Criterion Based on Accuracy (Rule 1)	First Selection Result	Second Selection Criterion Based on Time (Seconds)	Second Selection Result
RMSE<0.03	EoS	None	None
0.03≤RMSE<0.05	EoS	Time<12	None
		12≤Time<24	None
		24≤Time<36	None
		36≤Time<48	None
		48≤Time<60	None
		60≤Time<72	None
		72≤Time<84	5to 8 fold EoS
		84≤Time	9 fold EoS
0.05≤RMSE<0.07	EoS, KRG, RSM	Time<12	None
		12≤Time<24	None
		24≤Time<36	None
		36≤Time<48	2 fold EoS and 2 fold KRG
		48≤Time<60	None
		60≤Time<72	3 to 4 folds EoS and 3 to 4 fold KRG
		72≤Time<84	5 to 8 folds EoS, 5 to 8 folds KRG and 5 to 9 fold RSM
		84≤Time	9 fold EoS, 9 fold KRG and 9 fold RSM
0.07≤RMSE<0.09	RSM, RBF	Time<12	None
		12≤Time<24	None
		24≤Time<36	None
		36≤Time<48	None
		48≤Time<60	None
		60≤Time<72	4 fold RSM
		72≤Time<84	None

First Selection Criterion Based on Accuracy (Rule 1)	First Selection Result	Second Selection Criterion Based on Time (Seconds)	Second Selection Result
		84≤Time	9 fold RBF
0.09≤RMSE<0.11	RSM, RBF	Time<12	None
		12≤Time<24	None
		24≤Time<36	None
		36≤Time<48	None
		48≤Time<60	None
		60≤Time<72	3 fold RSM and 4 fold RBF
		72≤Time<84	5-8 folds RBF
		84≤Time	None
0.11≤RMSE	RSM, RBF	Time<12	None
		12≤Time<24	None
		24≤Time<36	None
		36≤Time<48	2 fold RBF and 2 fold RSM
		48≤Time<60	None
		60≤Time<72	3 fold RBF
		72≤Time<84	None
		84≤Time	None

For the hot rod rolling problem, three tables as decision trees corresponding to the three outputs are developed to manage the selection among the four types of surrogates. As we have three characteristics for the problem, accuracy, size and time, and time is directly proportional to size, the selection of appropriate surrogates can be based on accuracy and size. In this work, accuracy is used as the criterion for the first selection and size is the criterion for the second selection. Specific guidance for the selection of surrogate models for output Y1 is shown in Table 6.11 as an example. There is no surrogate model which gives us RMSE<0.03, while EoS

can be used if accuracy $0.03 \leq RMSE < 0.05$ is needed and here, EoS is the choice based on the first selection criteria (RMSE). Also, for the second selection criteria (time), 5-8 folds can be chosen when we need to spend no more than 84 seconds to run the code for the surrogate model.

6.5 On Verification and Validation

6.5.1. Empirical structural validation

Empirical structural validation involves accepting the appropriateness of the example problems used to verify the performance of the framework and the method. The example problem used in Chapter 6 is a hot rolling process chain. Using the hot rod rolling example problem in Chapter 6, the utility of the ensemble of surrogate creation framework and the method is tested for the enhancement in accuracy, speed, and computational complexity management. A discussion on the specific problem (hot rolling process chain) is carried out in detail. A literature review on hot rod rolling process is carried out. In addition to the validation of design methods, the chapter is also crucial from the standpoint of the major theme addressed in this dissertation. In this chapter, we discuss the validation of the proposed systematic method of ensemble of surrogate model creation.

6.5.2. Empirical performance validation

Empirical performance validation consists of accepting the usefulness of the outcome with respect to the initial purpose and accepting that the achieved usefulness is related to applying the framework and method. The utility of the proposed method is demonstrated by carrying out the ensemble of surrogate modeling in Chapter 6. In Chapter 6, the design architecture in terms of Gap 2, “creating ensemble of surrogates (EOS) which are more accurate than individual surrogates” is identified and described in detail after a critical evaluation of the literature.

To address this gap, Research Question 2, which is "What is the mathematics to create EOS that are more accurate than individual surrogate models?" developed in Chapter 6 of the dissertation. To answer the Research Question 2, Hypothesis 2, "Using a weighted average of individual surrogate models and minimum overall cross validation error, we can build such EOS" is posed. In this chapter, the industry inspired problem of focus in this dissertation is addressed. The key outcome of this chapter is a method to build an EOS that is both accurate and relatively less computationally expensive.

6.6 Closing Remarks on Building Ensemble of Surrogate Models

Based on the published literature, creating an ensemble of surrogates using cross-validation errors results is higher accurate but it is more computationally intensive than using individual surrogates. Our contribution in this chapter is to propose a method to build an ensemble of surrogates that is both accurate and less computationally expensive. The novelty in this chapter is to propose a method based on cross validation to find an ensemble of surrogates which is created by the least possible number of data points. The resulting ensemble surrogate has higher accuracy than each individual surrogate and is less computationally intensive. To achieve this ensemble surrogate, we compare it with individual surrogate models based on computation time, size and desired accuracy. For this purpose, we use RMSE as the accuracy measure, time of simulation as the computation performance measure, and the number of data points as the dimension measure. In Summary, we find that (1) it is effective to use cross-validation to study the impact of the size of the sample data set; (2) the highest accuracy with least required data and less computation time is achievable using the right number of samples; (3) an example of surrogates is relatively insensitive to the size of the sample data or number of data points. We create rule-based guidance for selection of surrogate models for one of the response variables in Table 6.11. Creating rule-

based templates for each response variable and putting them together into a knowledge-based platform and ontology is possible future research. As another possible research direction, these knowledge-based platforms and ontology can be used in automating the surrogate modeling process.

This chapter is based on several assumptions (1) hot rod rolling process is a single independent engineering process (2) we are able to create a surrogate model for hot rod rolling considering it as an independent engineering process. Removing these assumptions and considering the casting and reheating as preprocesses along with cooling and forging as post processes is another possible future research direction that we can take. Furthermore, we create rule-based guidance for selection of surrogate models for one of the response variables in Table 11. Creating rule-based templates for each and every response variable and putting them together into a knowledge-based platform and ontology is possible future research. As another possible research direction, these knowledge-based platforms and ontology can be used in automating the surrogate modeling process.

CHAPTER 7 CREATING AN AUTOMATIC MULTI-LEVEL SURROGATE MODEL SELECTION PROCESS

As shown in Figure 7.1, in this chapter, the foundations for the Gap 3 and automating the process of surrogate modeling is laid out.

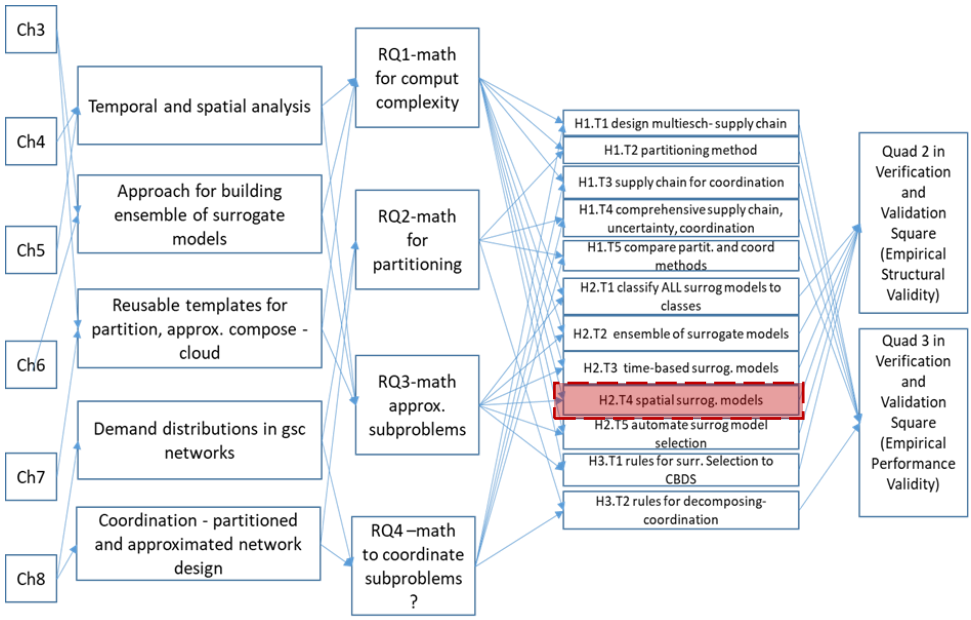


Figure 7.1. Relationship of research efforts with the temporal surrogate models and connection between chapters of the dissertation.

The relationship of these research efforts reviewed in this chapter with the constructs of the systematic approach developed in this dissertation is highlighted in Figure 7.1. in Section 7.1, a critical evaluation of the literature on surrogate model selection process is provided. While in Section 7.1.1, the manual comparison based surrogate model selection is described, the evolutionary algorithm based surrogate model selection is described in Section 7.1.2. Then, in Section 7.2, the multi-layer surrogate models' selection process proposed in this dissertation is described and performance measures are explained. In Section 7.3 the details of the proposed framework are provided followed by the verification of the proposed framework in Section 7.4. Then, the obtained results of demonstrating the utility of the proposed framework on several

examples are shown in Section 7.5. verification and validation of the chapter and closing remarks are finally provided in Sections 7.6 and 7.7, respectively.

Furthermore, the bigger picture of surrogate modeling automation problem and how it can be done in smaller operational levels are described. Details about how to create reusable templates and how to replicate and reuse them in a knowledge-based platform called cloud-based design system (CBDS) is provided. Then, step by step process of automating surrogate modeling process through a multi-level surrogate model selection and design process. A literature review on automated statistical modeling process through machine learning is carried out. Traditional manual selection approaches are very time-consuming and cannot be generalized.

To address these challenges, an evolutionary algorithm (EA)-based approaches are proposed and studied. However, they lack interpretability and are computationally expensive. To address these gaps, we create a rule-based method for an automatic surrogate model selection called AutoSM. The drastic increase in the selection pace by pre-screening of surrogate model types based on selection rule extraction is the scientific contribution of our proposed method. First, an interpretable decision tree is built to map four critical features, including problem scale, noise, size of sample and nonlinearity, to the types of surrogate model and select the promising surrogate model; then, a genetic algorithm (GA) is used to find the appropriate hyper-parameters for each selected surrogate model.

The AutoSM is tested with three theoretical problems and two engineering problems, including a hot rod rolling and a blowpipe design problem. According to the empirical results, using the proposed AutoSM, we can find the promising surrogate model and associated hyper-parameter in 9 times less than other automatic selection approaches such as concurrent surrogate model selection (COSMOS) while maintaining the same accuracy and robustness in surrogate model

selection. Besides, the proposed AutoSM, unlike previous EA-based automatic surrogate model selection methods, is not a black box and is interpretable. In addition to the validation of design method for automated surrogate model selection, the chapter is also crucial from the standpoint of the major theme addressed in this dissertation. In this chapter, we discuss the validation of the proposed systematic method of surrogate modeling automation.

Surrogate models have been widely used in engineering design because of their capability to approximate computationally complex engineering systems. In practice, the choice of surrogate models is extremely important since there are many types of surrogate models, and they also have different hyper-parameters. Traditional manual selection approaches are very time-consuming and cannot be generalized. To address these challenges, an evolutionary algorithm (EA)-based approaches are proposed and studied. However, they lack interpretability and are computationally expensive. To address these gaps, we create a rule-based method for an automatic surrogate model selection called AutoSM.

The drastic increase in the selection pace by pre-screening of surrogate model types based on selection rule extraction is the scientific contribution of our proposed method. First, an interpretable decision tree is built to map four critical features, including problem scale, noise, size of sample and nonlinearity, to the types of surrogate model and select the promising surrogate model; then, a genetic algorithm (GA) is used to find the appropriate hyper-parameters for each selected surrogate model. The AutoSM is tested with three theoretical problems and two engineering problems, including a hot rod rolling and a blowpipe design problem.

According to the empirical results, using the proposed AutoSM, we can find the promising surrogate model and associated hyper-parameter in 9 times less than other automatic selection approaches such as concurrent surrogate model selection (COSMOS) while maintaining the same

accuracy and robustness in surrogate model selection. Besides, the proposed AutoSM, unlike previous EA-based automatic surrogate model selection methods, is not a black box and is interpretable.

Glossary

CART	Classification and regression tree
CFD	Computational Fluid Dynamics
COSMOS	Concurrent Surrogate Model Selection
DoE	Design of Experiments
EA	Evolutionary Algorithm
EGO	Efficient Global Optimization
GA	Genetic Algorithm
KRG	Kriging
MARS	Multivariate Adaptive Regression Splines
MSPSO	Multiple Surrogates based Particle Swarm Optimization
NSGA-II	Non-Dominated Sorting Genetic Algorithm II
PRM	Polynomial Regression Method
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
RSM	Response Surface Models

7.1 Frame of Reference of Surrogate Model Selection Process

Surrogate models, also so-called metamodels, models of models, or response surface models (according to the context and research society), are statistical models that are used to replace computationally costly simulations and a rigorous technique to obtain empirical models from

physical experiments (Alizadeh et al., 2020a). They are considered as data-driven or predictive models that are created based on the design of (physical or numerical) experiments (DOE) (Qian et al., 2006). They are extensively used in engineering design as estimations of the analysis codes and simulations, which are less expensive to run, and produce understanding between inputs as design variables (vector X) and responses (vector Y).

For instance, in an engineering design problem to find the optimal air foil shape for an aircraft wing, an engineer simulates the airflow around the wing for different shape variables (such as length, curvature, and material). For many real-world problems, however, a single simulation can take many minutes, hours, or even days to complete. As a result, routine tasks such as design optimization, design space exploration, sensitivity analysis, and what-if analysis become impossible since they require thousands or even millions of simulation evaluations. One way of alleviating this burden is by constructing approximation models, known as surrogate models, response surface models, metamodels, or emulators, that mimic the behavior of the simulation model as closely as possible while being computationally cheap(er) to evaluate. In the aircraft wing design problem, different shape variables are the inputs (shown by x_i) for the surrogate model.

Also, each characteristic of the final produced wing, such as weight, strength, stiffness, bending, strain, and elasticity, are the response variables (y_j) which we want to measure. However, they are easily directly measured, or the cost of physical experiment is high, or the simulation time is too long and computationally expensive. Thus, let the original relationship between the set of input variables X and the collection of response variables Y is $Y = f(X)$, the surrogate model of this relationship would be $\hat{Y} = h(X)$ and $Y = \hat{Y} + \mathcal{E}$, such that ε denotes both the measurement and

estimation errors. The surrogate modeling process is to implement DOE to find an efficient set of computer simulations and use regression modeling to build an estimation of the simulation.

Afterward, the resulted estimations substitute the current simulation code. Surrogate models bring values of (i) realizing how input (X) and output (Y) variables are related; (ii) facilitating the process of combining simulation codes from different fields; (iii) providing a rapid computational method for design space exploration and optimization by building estimating based on costly simulations (Simpson et al., 2001). Surrogate models are essential in engineering design, nourishing a growing demand for fast and reliable models. However, the selection of surrogate models is always limited by the lack of prior knowledge (Mehmani et al., 2018), and existing surrogate model selection approaches are typically based on manual comparison and intuitive selection. Since the number of the existing types of the surrogate model has increased dramatically, manual comparison approaches become inefficient and unreasonable. Therefore, automated surrogate model selection methods have been proposed.

Most of the automated surrogate model selections are following Evolutionary Algorithms (EA). For example, the Genetic Algorithm (GA) is one of the most common EA, which is used to calculate the hyper-parameters and indirectly determine the choice of the optimization surrogate model type (Gorissen et al., 2010; Mehmani et al., 2018). Also, the extension of the genetic algorithm, such as the Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) is used to solve the multi-objective hyper-parameter optimization and the development of the surrogate model in (Couckuyt et al., 2011; Di Francescomarino et al., 2018; Passos and Luersen, 2018). Another example is the particle swarm optimization (PSO) algorithm, which is used in hyperparameter optimization (Gorissen et al., 2010; Toal et al., 2011). All existing automated surrogate model selection methods combine the selection process with various EA to automate the selection

process. The advantage of EA is its ability to handle a massive number of variables and high-dimensional problems. However, due to the complexity of the surrogate model selection problem and the trial-and-error mechanism of the evolutionary algorithm, the speed is still not satisfactory, which is the disadvantage of EAs.

To fill these gaps, an automated surrogate model selection method called AutoSM is developed in this chapter. The core idea behind AutoSM is to add knowledge of surrogate model selection to the process of automatic surrogate model selection. A set of selection rules created by the interpretable classification and regression tree (CART) method can pre-screen surrogate models and reduce the number of candidate surrogate model types. The knowledge of the surrogate model selection is always implicit and difficult to obtain; as such, the interpretable CART method enables us to extract the selection knowledge from benchmark problems (Gómez-Chova et al., 2003). Naturally, the CART model is composed of a series of "if-then" rules, which can be easily extracted and applied to the surrogate model selection process (Dandge and Chakraborty, 2020). In this chapter, these rules map four critical features of engineering problems, including problem scale, noise, size of the sample, and nonlinearity, to the types of surrogate models, which can filter out a large number of surrogate models and significantly reduce the candidates for surrogate models. After that, the Genetic Algorithm as an optimization algorithm is used to optimize the hyper-parameters and complete the entire surrogate model selection process. The performance of AutoSM is validated by its use for three theoretical problems and two practical design problems.

The scientific contribution of this chapter is automating the process of surrogate model selection. Also, embedding all four data characteristics of problem size, simulation time, non-linearity, and the number of samples in the proposed approach is another scientific contribution that enables the model to be as general as possible. Besides generality, interpretability is another

salient feature of the proposed approach, which together would allow designers to trace the surrogate model selection process and apply their domain knowledge wherever needed very quickly. This is another scientific contribution of the proposed approach, AutoSM. The last but not least scientific contribution is considering robustness in addition to the accuracy, which adds flexibility to the pre-surrogate selection process, where the designer models the design problem. In other words, the robustness or insensitivity of the selected surrogate model enables the designer to model the design problem in a more flexible, generic, and even less accurate manner.

The remainder of the chapter is organized as follows. First, the related work to the manual comparison based and EA-based surrogate model selection methods are presented in Section 7.2. Then, an introduction to the surrogate model selection problem is presented in Section 7.3, and an automated surrogate model selection method (AutoSM) is introduced in Section 7.4. In Section 7.5, the surrogate model pool and problem statements that contain different theoretical and application problems are given. The validation of the proposed approach, AutoSM, is shown in Section 7.6. Finally, we provide in Section 7.7 some closing remarks.

7.1.1. Manual comparison based surrogate model selection

Designing complex engineered systems often involving exploring the entire design space. This process needs constant running of simulations, which are computationally expensive. To decrease the computational expense of these simulations, the surrogate modeling process is created to approximate them; however, choosing the appropriate surrogate model itself is typically nontrivial. Surrogate model selection is often completed through a manual comparison for different engineering problems.

To facilitate an efficient understanding of different surrogate model structures and their suitable application characteristics, many review papers have been written. Simpson et al. suggest

that a designer's goal is usually to arrive at improved or robust solutions. Based on these goals, they generate recommendations for applying different techniques in given situations (Simpson et al., 2001). Wang et al. review and create an overview of surrogate modeling (metamodeling) techniques and their application in engineering design optimization (Wang and Shan, 2006). Razavi et al. review and categorize research efforts on surrogate modeling and applications, especially research conducted in the area of water resources, present a surrogate analysis framework (Razavi et al., 2012). Alizadeh et al. review more than 200 papers on surrogate modeling to classify the surrogate model selection process based on three main criteria, time, size, and accuracy. They also provide a qualitative relationship between the trade-offs among these three criteria (Alizadeh et al., 2020a).

More research is focused on using or developing new surrogate models to solve specific engineering problems or benchmark functions. Fitz et al. propose a surrogate model for cyber-physical systems to adequately explain cyber-physical systems and map them into the industry foundation classes (Fitz et al., 2019). Torabi et al. consider an aluminum blade in the process of forging and propose a surrogate response surface method (RSM) integrated with NSGA-II to optimize the performance of the forging process (Torabi et al., 2017). Denkena et al. create a new technique for material identification in Computer Numerical Control (CNC) (Denkena et al., 2018). Wang et al. find the Gaussian process as the best surrogate model construction method of the six candidate surrogate models for the Hartman function (Wang et al., 2014). Li et al. propose a two-level multi-surrogate assisted optimization (TMAO) to decompose high dimensional problems and obtain stable and accurate results for small or medium-sized problems (Li et al., 2016b). Lelièvre et al. develop a method called the active learning reliability method, which is a

combination of Kriging (KRG) and Monte Carlo simulations, to deal with small failure probabilities and time-consuming models (Lelièvre et al., 2018).

Another approach to study the surrogate model selection process is to identify general rules. For example, Williams and Cremaschi conduct several known benchmarks using eight surrogate models to find a suitable surrogate model based on merely the features of the benchmark dataset (Williams and Cremaschi, 2019). However, they do not consider the problem scale, the type of simulation (random or deterministic), as well as the demands (accuracy and robustness). To test the effectiveness of various approaches for different classes of problems, Jin et al. select fourteen test problems and classify them based on three representative features of engineering design problems: 1. Problem scale (the number of input variables), 2. Nonlinearity (low or high), and 3.

Noisy or smooth (Jin et al., 2001). Garbo and German proposed an approach to supervising the sampling phase with a surrogate model built from the available training set to make its behavior reliable for the selected SM formulation (Garbo and German, 2019). Zhou et al. build a multi-fidelity metamodeling framework to automate the process of collecting the fidelity knowledge as prior-knowledge for the metamodeling process (Zhou et al., 2016). Then, the same group created an active learning surrogate modeling method by consecutively deriving the knowledge from variable-fidelity models (Zhou et al., 2017). Mehmani et al. present a multi-level surrogate selection process using metrics like the median predictive estimation of model fidelity, maximum and median error on four benchmark functions, and three real-world problems (Mehmani et al., 2018).

Some other studies studied different scenarios generated by surrogate models (Alizadeh et al., 2016b; Alizadeh et al., 2020c; Sadaghiani et al., 2014) models. These papers consider the features of the benchmark dataset; however, they do not use dependable measures of model robustness,

and we add in this chapter. The ontology-based method is the way to use the derived rules to automate the design process (Bock et al., 2010; Gehlert and Esswein, 2007; Zaletelj et al., 2018; Zhou et al., 2017). To create an ontology, we need prior rule-based knowledge to create knowledge-based reusable templates. To generate these knowledge-based templates, automatic selection methods are required to create the necessary rules. An evolutionary algorithm is one way to automate the surrogate model selection, which is explained in the next section.

7.1.2. Evolutionary algorithm based surrogate model selection

Yu et al. compare the quality indices to select the most promising model in the surrogate-assisted evolutionary algorithm. They also investigated the compatibility among accuracy and ranking refinement approaches (Yu et al., 2019). Gorissen et al. propose an automated two-step surrogate model selection process. Genetic algorithms (GAs) are used to calculate the hyper-parameters, whose values indirectly determine the choice of the optimization model type.

Mehmani et al. propose an automated surrogate model selection framework called concurrent surrogate model selection (COSMOS), which involves a genetic algorithm in the performance of the optimal choice (Mehmani et al., 2018). A Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) is used to solve the multi-objective hyper-parameter optimization and automate the surrogate model development in (Di Francescomarino et al., 2018). Couckuyt et al. extend the well-known efficient global optimization (EGO) method with an automatic surrogate model type selection framework that can dynamically select the best model type depending on the available data (Couckuyt et al., 2011).

Nguyen et al. propose multiple surrogates based particle swarm optimization (MSPSO) framework (Lv et al., 2019). Cicirelli et al. create a smart, automated surrogate model framework to converge two different perspectives, including functional design specifications and data

structures (Cicirelli et al., 2017). Díaz-Manríquez et al. present an evolutionary algorithm to automate the surrogate modeling process and compare the efficiency of these surrogate models using accuracy and precision measures (Díaz-Manríquez et al., 2017). Based on this critical evaluation of the literature, a summary of surrogate model selection literature is shown in

Table 7.1.

Table 7.1. Critical evaluation of the literature and gap identification.

	Characteristic of Data				**Demands**		**Automation level**	
Features	Problem scale	Simulation type	Non-linearity	No. of samples	Accuracy	Robust	Manual	Automated
Paper								
Simpson et al., 2001 (Simpson et al., 2001)	√	deterministic			√	√	√	
Jin et al., 2001 (Jin et al., 2001)	√	deterministic	√		√		√	
Razavi et al., 2012 (Razavi et al., 2012)	√	random		√	√		√	
Wang and Shan, 2006 (Wang and Shan, 2006)	√	random		√	√		√	
Gorissen et al., 2009 (Gorissen et al., 2010)		deterministic			√			√

Features	Characteristic of Data				Demands		Automation level	
	Problem scale	Simulation type	Non-linearity	No. of samples	Accuracy	Robust	Manual	Automated
Paper								
Wang et al., 2014 (Wang et al., 2014)		random		√	√	√	√	
Lelievre et al., 2018 (Lelièvre et al., 2018)	√	random			√	√	√	
Shafiei et al., 2016 (Shafiei Kaleibari et al., 2016)	√	random			√		√	
Lv et al., 2019 (Lv et al., 2019)		random	√		√			√
Simpson et al., 2008 (Alizadeh et al., 2016c)	√	random		√	√		√	
Couckuyt et al., 2011(Simpson et al., 2008)		deterministic		√	√		√	
Wang et al., 2014 (Couckuyt et al., 2011)	√	deterministic			√			√
Di Francescomarino et al., 2018 (Di	√	random			√	√		√

Features	Characteristic of Data				Demands		Automation level	
	Problem scale	Simulation type	Non-linearity	No. of samples	Accuracy	Robust	Manual	Automated
Paper								
Francescomarino et al., 2018)								
Viana and Haftka, 2008 (Viana and Haftka, 2008)	√	deterministic		√	√		√	
Cicirelli et al., 2017 (Cicirelli et al., 2017)	√	deterministic			√			√
Viana et al., 2009 (Viana et al., 2009)	√	deterministic		√	√		√	
Viana et al., 2018 (Viana et al., 2018)	√	random		√	√		√	
Williams and Cremaschi, 2019 (Williams and Cremaschi, 2019)		random	√	√	√		√	
Yu et al., 2019 (Yu et al., 2019)	√	random	√	√	√		√	
Zhou et al., 2017 (Zhou et al., 2017)				√	√		√	

Features	Characteristic of Data				Demands		Automation level	
	Problem scale	Simulation type	Non-linearity	No. of samples	Accuracy	Robust	Manual	Automated
Paper								
Mehmani et al., 2018 (Mehmani et al., 2018)		random	√	√		√		√
Diaz-Manriquez et al., 2017 (Díaz-Manríquez et al., 2017)		deteministic		√		√	√	
Singaravel et al., 2019 (Singaravel et al., 2019)		random	√	√	√			√
Wang et al., 2018 (Wang et al., 2018)		random	√	√		√	√	
Fitz et al., 2019 (Fitz et al., 2019)		random	√	√	√		√	
Our work	√	both	√	√	√	√		√

Notably, there is a lack of a comprehensive surrogate model selection tool to fully consider the features of the data and requirements for engineering problems. Also, due to the gigantic trial-and-error iterations of evolutionary algorithms, all existing EA-based selection methods are computationally expensive. To address this gap, we create a rapid, automatic, interpretable surrogate selection method, which includes all the requirements of the features listed in

Table 7.1. Therefore, an interpretable, automated surrogate model selection approach based on the criteria of problem size, simulation time, non-linearity, and the number of samples is needed, which aims at optimizing the robustness and accuracy of the chosen surrogate model.

7.2 Surrogate Model Selection Problem

To address the surrogate model selection problem, with each selection approach, we must answer two critical questions: (1) what must be done in the surrogate model selection process and (2) how to determine the best choice. The corresponding answers are provided by the *structure of surrogate model selection* (Section 7.3.1) and *performance metrics* (Section 7.3.2).

7.2.1. Three-layer structure of surrogate model selection

For a complete description of the surrogate model selection process, we refer to (Mehmani et al., 2018) and present a three-layer model selection structure. An illustration of the surrogate model selection structure is shown in Figure 7.22, which contains three components as follows:

i) Surrogate types: for example, KRG, radial basis function (RBF), RSM, and multivariate adaptive regression splines (MARS);
ii) Optional types: such as the covariance functions of KRG; for example, the Linear, Gaussian and Power covariance functions;
iii) Hyper-parameters for each optional type: for example, the correlation function parameter θ of the Gaussian covariance function of KRG. (More detail can and be found in Table 7.2)

Algorithm 1 General Framework *Sn*, surrogate type number; *On*, optional type number; ***S***, surrogate types; ***O***, optional types. *HP, hyper-parameter*

Input: *data*

Output: *satisfied surrogate optional types (S-O) & Hyper-parameters*

for i=1 to *Sn* **do:**

/*Layer1 Surrogate types selection*/

Surrogate type (*st*) = ***S(i)***

for j=1 to *On* **do:**

/*Layer2 Optional types selection*/

Optional type (*ot*) = ***O(j)***

while *not end of termination* **do**:

/*Layer3 Hyper-parameter optimization*/

Error = **FitnessFunction** (*st*, *ot*)

HP_optimal = **Optimization** (Error, HP)

[*minError, index*] = **min**(Error)

satified $S-O = S-O_{index}$

end for

end for

Key innovations

Figure 7.2. Three-layer structure of the surrogate model selection process.

Generally, the selection of the surrogate type and optional type relies on the optimization result of hyper-parameters, so the surrogate and optional type are always combined to make the comparison. An illustration of the application of the three-layer surrogate model selection structure is presented in Figure 7.2. In layer one and layer 2, surrogate types and their corresponding optional types are listed and directly compared, after the individual hyper-parameter optimization of each surrogate-optional type in layer 3. Evolutionary algorithms associated with typical evaluation functions are always used to optimize the hyper-parameter. The optimal hyperparameter is

developed through a large number of 'trial and error' iteration. Finally, satisfied surrogate-optional $S-O$ types and their optimal hyper-parameters (*HP_optimal*) are developed. It can be concluded that the surrogate model selection approach has to solve two key problems: 1. Surrogate type and optional type selection; 2. Surrogate model hyper-parameters selection (optimization). By the way, our major innovation happens at Layer 1 and Layer 2, which generates a selector to pre-screen surrogate models to speed up the selection of surrogate model.

7.2.2. Metrics for performance measures

Accuracy and robustness are regarded as the most critical requirements for suitable surrogate models in the engineering design; as such, two metrics are proposed to measure the accuracy and robustness in this work. Accuracy is a description of the systematic error, a measure of the difference between the predicted value of the model and the actual value (Khoshelham, 2011). R-squared (R^2) is a systemic error measure, which tells how good the best fit line is from the baseline model. So, it is the percentage number whose value is between 0 and 1. As such, comparing with other accuracy metrics, R^2 has the advantage of the property that its scale is intuitive. It ranges from 0 to 1, with 0 indicating that the proposed model does not improve prediction over the mean model, and 1 indicating perfect prediction. Therefore, we choose R^2 to measure the accuracy of our work.

$$R^2=1-\sum_{i=1}^{N_{test}}\left(y_i-\widehat{y_i}\right)^2\Big/\sum_{i=1}^{N_{test}}\left(y_i-\overline{y_i}\right)^2 \qquad \text{Equation 7.1}$$

where y_i is the observed values, and $\overline{y_i}$ denotes the mean of the observed values. N_{test} is the number of testing data.

Robustness is the characteristic of a surrogate model to remain unaffected by small variations in parameters (Wieland and Wallenburg, 2012). Robustness for the surrogate model describes its anti-interference ability. In other words, if the surrogate model can keep its prediction accuracy for different problems, it has good robustness. Therefore, based on the calculation of R^2 for accuracy, we further produce the calculation of the robustness.

$$rb=\sum_{i=1}^{M}\left(R_i^2-\bar{R}^2\right)^2\Big/M \qquad \text{Equation 7.2}$$

Where rb stands for the robustness, R_i^2 represents the R-squared value of the surrogate model in i^{th} benchmark problem, and $\bar{R}^2$ is the average R^2 value of all conditions. M denotes the number of different benchmark problems. The closer rb to 0, the more robust the surrogate model is.

7.3 AUTOSM Framework

The model type selection and model parameter selection are carried out by the surrogate model selection method. The primary work of this chapter is to generate a selector to pre-screen surrogate models and reduce the number of candidate surrogate model types. We extract four parameters to give a unified standard description to all engineering problems: then, building the mapping relation between these four parameters with the best metamodels and their options. As such, relying on the mapping relation, several candidates (good) surrogate-optional types can be provided based on the mapping relation. More details, shown in Figure 7.3, the AutoSM is composed of two phases. In the **offline phase**, the selector, which mapping the four features with best surrogate model types, is trained. The training process involves three operators: the 'Feature Extractor' (Section 4.1), the 'Surrogate model evaluator' (Section 7.4.2) are used to generate the training data, and the 'Model type selector' (Section 7.4.3) is used to develop the surrogate model type selector and selection

knowledge. Then in the **online phase**, the first step is the 'feature extract' for the new problem. With the specific value of those four features, they are entered into the selector to predict the candidate surrogate-optional types. By the way, the selector filters out many surrogate models and significantly reduces the number of candidate surrogate model types. Then, as the model types have been confirmed, their hyper-parameters also need to be identified since they also have a significant impact on the performance of surrogate models. GA is implemented as the hyper-parameter calculator, and its implementation can be found in a Matlab toolbox. http://www.sumo.intec.ugent.be/. After identifying the surrogate-optional types and hyper-parameters, the promising model types can be finally confirmed through the performance comparison between those candidate model types.

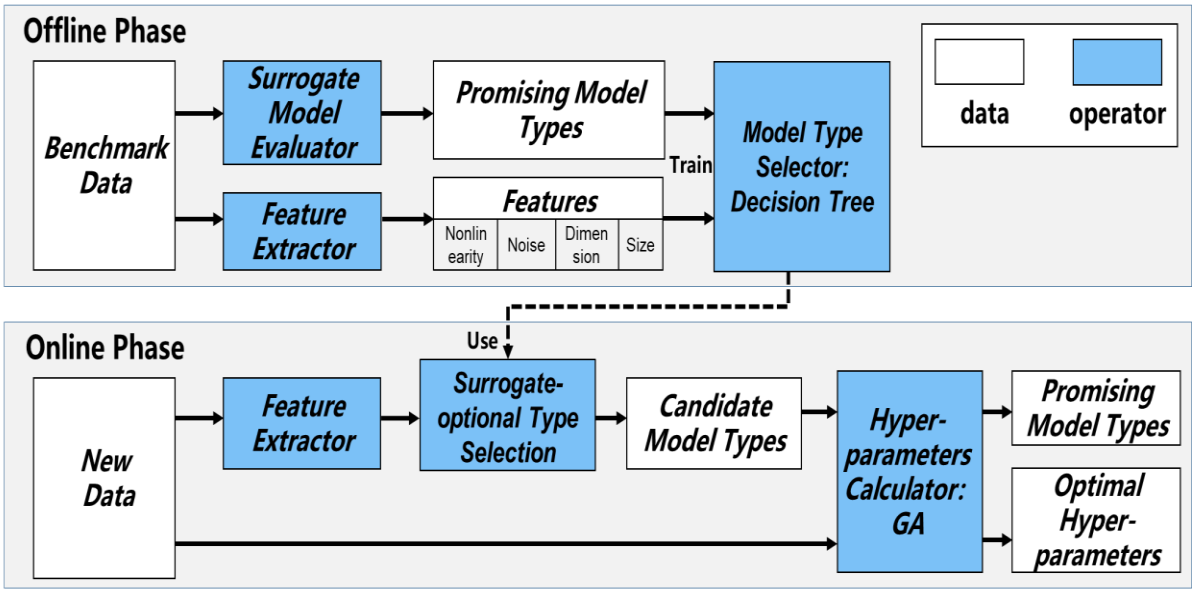


Figure 7.3. AutoSM framework consists of two phases, the offline phase is used for the selector generation, and an online phase is used for the new data surrogate model type prediction

7.3.1. Feature extractor

To have a unified standard to describe different datasets, we evaluate a set of meta-features for each benchmark dataset $\boldsymbol{D}$. We define meta-features as common features in all the engineering

problems. Different from mathematical problems, engineering problems have their meta-features, and their datasets should also reflect their meta-features. The selected meta-features are listed as follows:

1) Problem scale

Problem scale denotes the input variable dimension or the number of variables. Since AutoSM is to be used for engineering problems whose input variable dimensions are not large, we set the '*Dimension'* as a categorical value from 1 to 10.

2) Size of sample

The size of the sample for a particular problem, *'Size',* is determined by Experimental Design (DoE), and it is also associated with the *Dimension'*.

3) Deterministic VS random simulation (Noise)

A deterministic simulation is a computer experiment, where the samples are generated without any noise, while the random simulation includes uncertainty and is closer to physical experiments (Wang and Shan, 2006). '*Noise'* is a Boolean data. In our study, the noise behavior is artificially created by using local variations of the benchmark functions.

4) The nonlinearity of the performance behavior

The linearity of the performance behavior can be expressed as the power value of the Pearson correlation coefficient when using first or second-order polynomial models (Jin et al., 2001). The equation of Linearity and Nonlinearity can be expressed as follows:

$$\text{Linearity}=\left(\frac{\text{cov}(Y_1,Y_2)}{\sqrt{D(Y_1)}\sqrt{D(Y_2)}}\right)^2 \quad \text{Equation 7.3}$$

$$\text{Linearity}=1-\left(\frac{\text{cov}(Y_1,Y_2)}{\sqrt{D(Y_1)}\sqrt{D(Y_2)}}\right)^2 \quad \text{Equation 7.4}$$

where Y_1 and Y_2 denote predicted and real response, respectively. $cov(Y_1, Y_2)$ represents the covariance of Y_1 and Y_2, while $D(Y_1)$ and $D(Y_2)$ are the variances of Y_1 and Y_2, respectively. Since the first-order polynomial model is the most suitable model for highly linear benchmark functions, we fit it to each benchmark function and calculate the nonlinearity value for them. '*Nonlinearity'* is used as a continuous numerical description of the complexity of the problem. The closer '*Nonlinearity'* to 1, the more complex the problem is.

7.3.2. Surrogate model evaluator

To train the model type selector, we need to find the promising surrogate model types (labeled as A^*) for each benchmark data, the promising surrogate model types are the selected types type in the first and second layers of the three-layer surrogate model selection structure. Then the extracted meta-features and developed label (A^*) can be integrated and used as the training data to develop the model selector; finally, the selector can be used to predict the promising surrogate model types for the new data through its generalization. We choose the COSMOS method from (Mehmani et al., 2018) to find the promising surrogate model types for benchmark data.

7.3.3. Model type selector

All benchmark datasets can be described as meta-features $(\boldsymbol{mf})$, and their corresponding surrogate model types are labeled as $(\boldsymbol{A})$. As such, we use a mapping model $g(\cdot)$ to identify the relation between meta-features and promising surrogate-optional types $(\boldsymbol{A})$. The mathematical definitions of model type selector are defined in Equation 7.5 to Equation 7.7.

$$S=\{(\mathbf{mf_1}, A_1),\ldots,(\mathbf{mf_m}, A_m)\} \quad \text{Equation 7.5}$$

$$S_{train}=\{(\mathbf{mf_1}, A_1),\ldots,(\mathbf{mf}_{p-1}, A_{p-1})\} \quad \text{Equation 7.6}$$

$$S_{valid}=\{(\mathbf{mf_p}, A_p),\ldots,(\mathbf{mf}_m, A_m)\} \quad \text{Equation 7.7}$$

Let S be the new dataset split into S_{train} and S_{valid}. For $i = 1,2 \ldots m$, let $\boldsymbol{mf_i} \in \mathbb{R}^{\boldsymbol{d}}$ denote a feature vector, $A_i \in \boldsymbol{A}$ is the corresponding model type. So, the prediction output achieved on $\boldsymbol{mf_p}, \ldots, \boldsymbol{mf_m}$ is represented by $\hat{A}_p, \widehat{\ldots, A_m}$ when S_{train} is used for training. In our work, Classification and Regression Tree (CART) is selected as the mapping function $g(\cdot)$.

Naturally, the interpretable CART model is composed of a series of "*if-then*" rules, which can be easily extracted and applied to the surrogate model selection process. Based on these rules, many surrogate model types can be filtered out, which can greatly speed up the surrogate model selection. The selection of candidate surrogate model types for a new dataset D_{new} is explained as in eq.8.

$$D_{new} \rightarrow mf^* \rightarrow g(mf^*) \rightarrow \hat{A} \quad \text{Equation 7.8}$$

where mf^* and $\hat{A}$ are the meta-feature and predicted surrogate-optional types of the new dataset D_{new}. First, meta-features of D_{new} are extracted and labeled as mf^*; then, the decision tree algorithm $g(mf^*)$ is used to identify candidate surrogate-optional types $\hat{A}$, which filters out a significant number of surrogate model types.

7.4 Verification of AUTOSM

7.4.1. Surrogate model pool

The surrogate model pool considered for selection in the version of AutoSM presented here follows the composition that includes four surrogate models i) RSM, ii) KRG, iii) RBF, iv) MARS. The surrogate-optional types and their corresponding hyper-parameters are listed in Table 7.2. Also, brief descriptions of the working principle of each surrogate model are provided in Appendix A. The program of this work is based on several Python-based implements called LinearRegression, pyKriging, RBF, Earth, respectively.

Table 7.2. Surrogate-optional types and their corresponding hyper-parameters in the surrogate model pool

	Surrogate-Optional Type	
Surrogate Type(S)	Optional Type(O)	Hyper-Parameters (HP)
RSM	Order=2	
	Order=3	
	Order=4	
	Order=5	
	Order=6	
KRG	Linear: $max(1-\theta r, 1)$	correlation function parameter θ
	Power: θ^r	correlation function parameter θ
	Gaussian: $e^{(\theta r)^2}$	correlation function parameter θ
	Exponential: $e^{-\theta r}$	correlation function parameter θ
	Spherical: $1-3\varphi^2+2\varphi^3;\ \varphi = max(1-\theta r, 1)$	correlation function parameter θ

Surrogate-Optional Type		
Surrogate Type(S)	Optional Type(O)	Hyper-Parameters (HP)
RBF	Multiquadric: $\left(1/\left(\frac{1}{r\sigma}\right)^2+1\right)^{\frac{1}{2}}$	shape parameter σ
	Cubic: r^3	—
	Inverse: $1/\sqrt{\left(\frac{1}{r\sigma}\right)^2+1}$	shape parameter σ
	Linear: r	—
	Thinplate: $r^2\log(r)$	—
MARS	Max_Interaction=2	—
	Max_Interaction=3	—
	Max_Interaction=4	—

7.4.2. Theoretical problems

There are three theoretical benchmark functions used in our AutoSM framework: 1) Perm function (Mehmani et al., 2018), 2) the Dixon and Price function (Wang et al., 2014), and 3) Beale function (Wang et al., 2014). These functions have been widely used for determining the most accurate surrogate model in other related research. The detail of these functions is given in Appendix B. To test the performance of AutoSM, we select different sample sizes of data from each theoretical problem to increase the size of the test database. For validation purposes, we create a broad set of additional test datasets for these problems and identify their corresponding best surrogate-optional type as listed in Table 7.3. Since the test data is obtained by sampling from the deterministic functions, the meta-feature '*Noise'* is set to zero here.

Table 7.3. Meta-features and the corresponding best surrogate optional type for theoretical problems

Function Name	Mete-features				Best Surrogate-Optional Type
	Nonlinearity	Dimension	Size	Noise	
Perm500	0.85	10	500	0	MARS_3
Perm40	0.99	10	40	0	RSM_3
Perm280	0.85	10	280	0	RSM_6
Perm160	0.90	10	160	0	RSM_4
Perm1000	0.82	10	1000	0	MARS_2
DixonPrice80	0.17	5	80	0	KRG_gaussian
DixonPrice500	0.11	5	500	0	RSM_4
DixonPrice400	0.12	5	400	0	RSM_4
DixonPrice300	0.12	5	300	0	RSM_4
DixonPrice200	0.13	5	200	0	RSM_4
DixonPrice20	0.72	5	20	0	RBF_multiquadric
Beala90	0.51	2	90	0	MARS_3
Beala70	0.60	2	70	0	MARS_2
Beala50	0.45	2	50	0	RBF_cubic
Beala30	0.58	2	30	0	MARS_2
Beala120	0.39	2	120	0	RSM_3
Beala10	0.47	2	10	0	RBF_inverse

We demonstrate the distribution of the test data in Figure 7.4, and it is evident that the test data covers most of the design space. So, the diversity of our test data is sufficient to validate the performance of AutoSM.

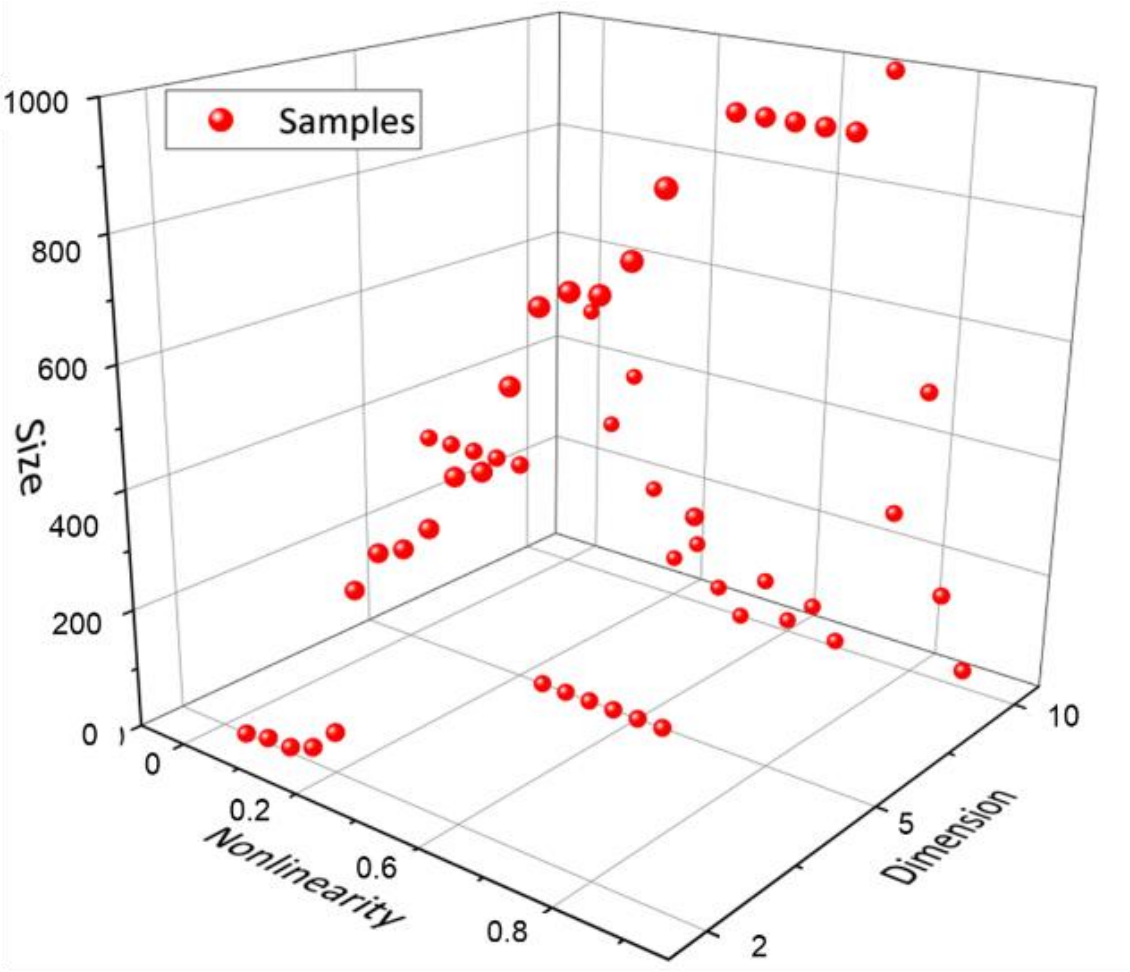


Figure 7.4. Distribution of the test data generated from theoretical problems

We demonstrate the distribution of the test data in Figure 7.4, and it is evident that the test data covers most of the design space. So, the diversity of our test data is sufficient to validate the performance of AutoSM.

7.4.3. Test problems used to demonstrate the utility of the proposed framework

Hot rod rolling and blowpipe design are two engineering problems that we have thoroughly studied. For the hot rod rolling problem, a simulation program (STRUCTURE developed by (Jones and Bhadeshia, 1997)) is used to study the structural performance of the final product. In order to optimize the construction of blowpipe to improve its reliability, time-consuming computational

fluid dynamics (CFD) simulations (Fluent) is used to identify the stress and plastic strain distribution of different blowpipe design case. Both of these two problems need computationally expensive simulations, which often take more than 24 hours for a single design case. Using surrogate models, the computational complexity of these problems is manageable. As such, the hot rod rolling, and blowpipe design problems are selected as our application problems to test the practicality and effectiveness of AutoSM.

Hot rod rolling

As shown in Figure 7.5, a hot rod rolling process to create a rod from a lab of steel. This is an intermediate process in the creation of an automotive steel gear. In this deformation process, the steel, which initially is in the austenite phase is converted to ferrite during slow cooling. On the other hand, in fast cooling, austenite is transformed to ferrite and from ferrite to pearlite. Also, to determine the final phase, there are three crucial process variables, which are the percentages of carbon and manganese in the alloy, the cooling rate, and the initial austenite grain size. For instance, low carbon percentage and small austenite grain size facilitate the phase transformation to ferrite. Also, the percentage of manganese causes an undesirable banded microstructure (Jägle, 2007). A banded microstructure results in fractures during any processing phase following hot rod rolling, so it is critical to avoid it. Therefore, we need to be able to predict the outcome of the hot rod rolling process precisely. These variables are defined in Figure 7.5. The output variables of the hot rod rolling process are pearlite, and two types of ferrite, namely allotriomorphic ferrite and Widmanstätten ferrite based on different values of each of four input variables. Therefore, the process has four input variables and three output variables (Alizadeh et al., 2019).

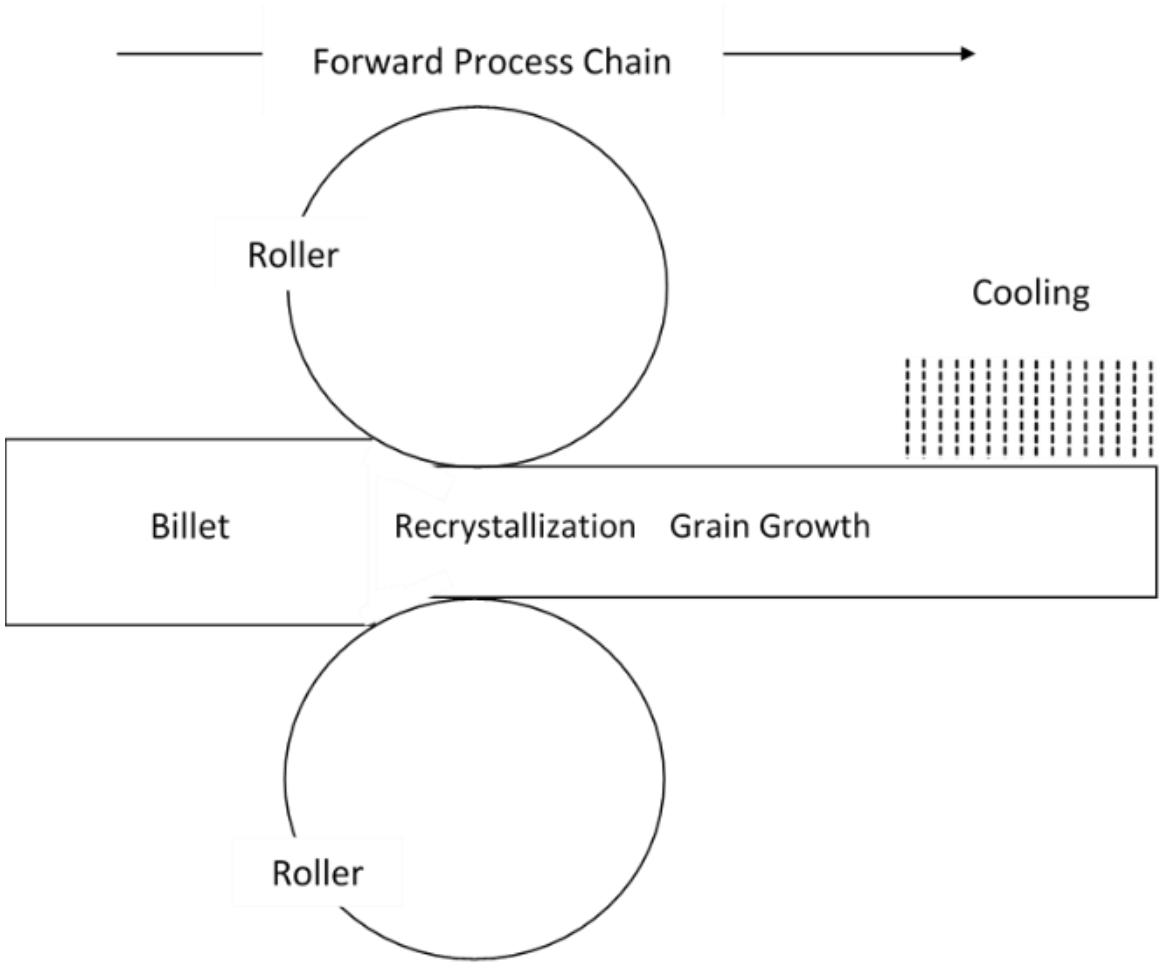


Figure 7.5. Hot rod rolling process

Nellippallil et al. in (Nellippallil et al., 2018), using the simulation program, STRUCTURE developed by (Jones and Bhadeshia, 1997) use a full factorial design of experiments to generate a data set and study how different values of four input variables, cooling rate, carbon concentration, manganese concentration, and austenite grain size, affect the final product which can be three phases (pearlite, allotriomorphic ferrite, and Widmanstätten ferrite). Nellipallil et al. 's work are verified by comparing the simulation predictions with experimentally measured data reported by (Bodnar and Hansen, 1994).

Table 7.4. Design variables and their definition of hot rod rolling design

Design Variable	Definition
Manganese concentration after rolling (Mn)	Jones et al. (Jones and Bhadeshia, 1997) point out that manganese is an austenite stabilizing agent, meaning the banded transformation is low in the regions where manganese concentration is high. In essence, this leads to an accumulation of carbon, which leads to the formation of pearlite.
Final Austenite grain size after rolling (AGS)	This parameter has an inbuilt effect on grain boundary area per unit volume and thus on nucleation itself. Because of this effect, and the simultaneous phase transformations, the average grain size (neglecting the length scale) has a significant bearing on the final microstructure.
Cooling rate	High cooling rates usually suppress banding. Lower cooling rates favor carbon diffusion leading to the development of banded microstructure.
Carbon content	The carbon content changes the physical properties of commercially available steel and hence determines which component is formed first during the initial stages of cooling

Blowpipe design

In the steelmaking industry, blowpipes are widely used as transfer devices to ensure a constant supply of hot air to the blast furnace, where it connects the bustle pipe to the furnace tuyeres. In order to optimize the construction of blowpipe to improve its reliability, computational fluid dynamics (CFD) was used to identify areas with high stress and plastic strain corresponding to different input conditions. However, to develop a satisfactory construction, a large number of

simulation experiments are required, which is very time-consuming and expensive. Typically, surrogate models are useful for dealing with such a problem.

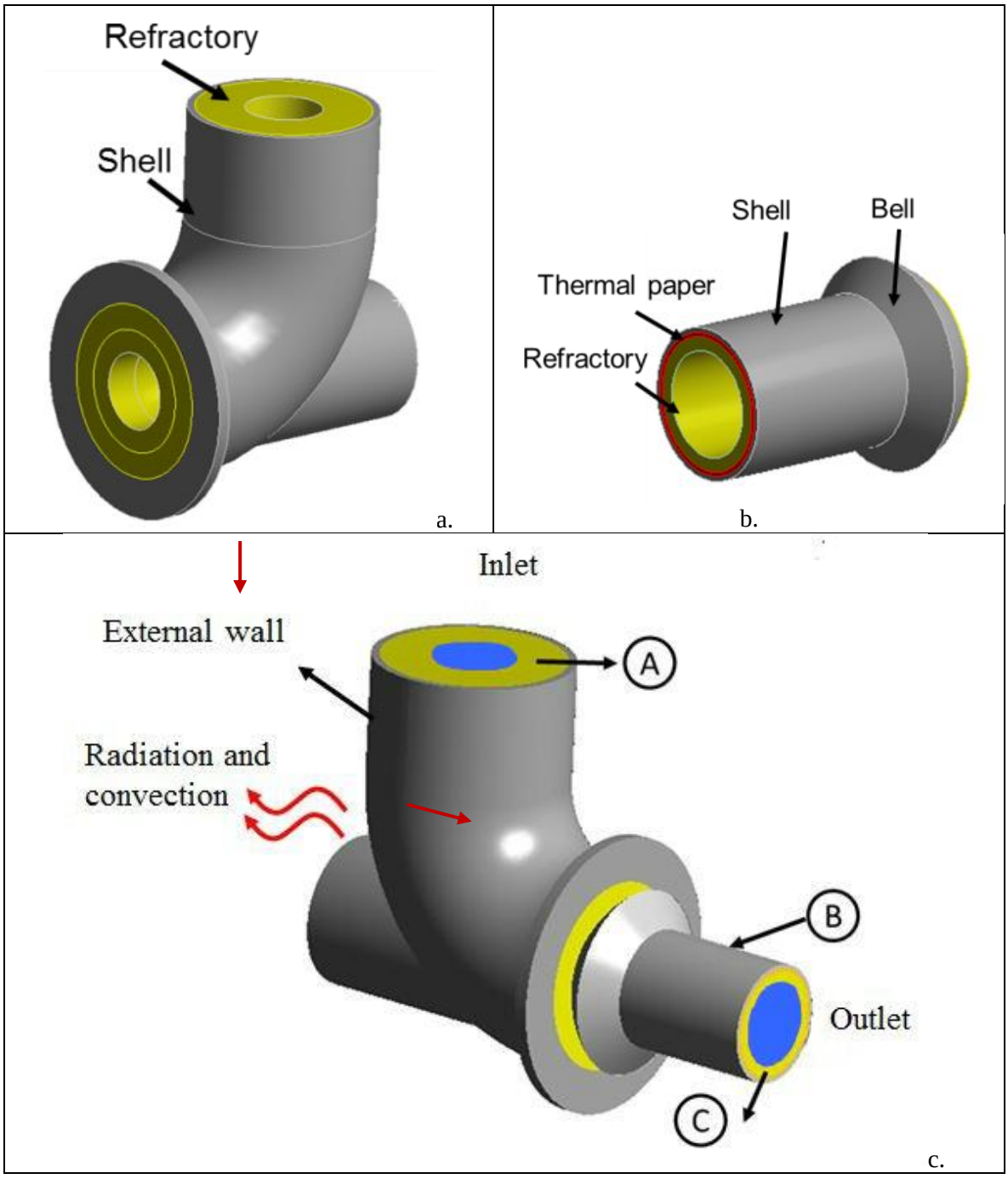


Figure 7.6. The geometry of elbow (a.), blowpipe (b.) and blowpipe system(c.) *(Cui et al., 2018).*

In Figure 7.6, we show the three-dimensional model of the blowpipe system. A complete blowpipe system consists of two components: elbow and blowpipe, as shown in Figure 7.6 (c.). The hot blast air passes from the inlet and moves from the elbow to blowpipe. In the whole process, the blowpipe is critically affected by two aspects: i) Hot Blast Temperature (°F), the hot and high-speed airflow which leads to a continuous high-temperature and high-pressure environment; ii) The components, including thermal (thickness (in.)) and length (in. from the bell)) and refractory (thickness (in.) and material position (in.)). The output responses are stress (psi) and temperature distributions (°F) in a blowpipe system, which can also be expressed as the stress and temperature at each location in the blowpipe. In Table 7.5, we show the meta-features and the best type of these two application problems. Since the test data is obtained from industrial problems, the meta-feature 'Noise' is set to 1 here.

Table 7.5. Meta-features and the corresponding best surrogate optional type for hot rod rolling and blowpipe design problems

Function Name	**Mete-features**				**Best Surrogate-Optional Type**
	Nonlinearity	Dimension	Size	Noise	
HRR_Y1	0.0298	4	20	1	KRG_gaussian
HRR_Y2	0.0915	4	20	1	KRG_gaussian
HRR_Y3	0.0104	4	20	1	KRG_power
Blowpipe_Y1	0.0992	5	334	1	RSM_6
Blowpipe_Y2	0.6744	5	209	1	KRG_power
Blowpipe_Y3	0.7296	5	209	1	KRG_power

7.5 Results and Discussion on Creating an Automatic Multi-Level Surrogate Model Selection Process

The entire experimental system for validating the accuracy and robustness of the proposed AutoSM is shown in Figure 7.7. Four different surrogate models with the entire 18 surrogate-optional types are employed. Two parts of the database, *Training Data* (from 10 benchmark functions) and *Test Data* (from 3 theoretical and two application problems), are developed to train, validate and test our selection model (shown in Appendix C). From Figure 7.7, there are two ways to calculate the accuracy of our selection model (CART): i) Using K-fold cross-validation to validate the accuracy by the validation data generated from 10 training benchmark functions; ii) Using test data.

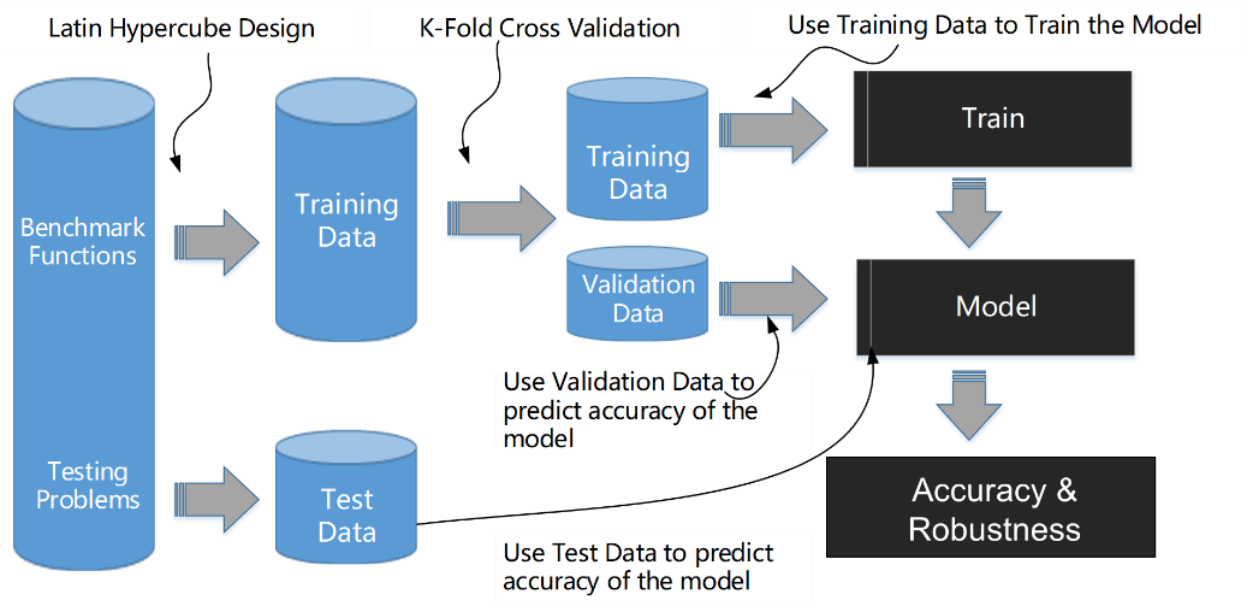


Figure 7.7. Experimental system for validating the accuracy and robustness of the selection model.

The average predicted accuracy for each benchmark function obtained by the AutoSM framework is listed in Table 7.6. The entire predicted accuracy of our selection model is the average value of all benchmarks (about 62%). More specifically, the selection model is most

accurate in predicting T5, T2, and HM6 benchmark functions, especially for T5 and T2 (about 95%).

Table 7.6. Average predicted accuracy for benchmark functions.

	Benchmark functions									
	T5	T4	T3	T2	T1	HM6	HM3	GP	CB	BH
Predicted Accuracy (%)	89.47	42.11	57.89	94.74	63.16	89.47	63.16	42.11	52.63	63.16

The recommended candidate surrogate-optional types (CSOTs) obtained by AutoSM for theoretical and application problems, and the label (Best Type) of each problem, are listed in Table 7.7. If the 'Best Type' is one of the three Candidate Surrogate Optional Types (CSOTs), it is deemed to be an accurate prediction (set to 1 in the table), otherwise set to 0. From Table 7.7, the validation accuracy can be easily calculated, about $\frac{18}{23} \approx 78\%$, where the validation accuracy of the application and theoretical problems are $\frac{5}{6} \approx 83\%$ and $\frac{13}{17} \approx 76\%$, respectively.

Table 7.7. The set of promising surrogate models (if the Best Type is one of the three CSOTs, bold the corresponding item)

Problem		Promising Surrogate Models		Accurate prediction
		Best Type	Candidate Surrogate_Optional Types (CSOTs)	
Application Problems	HRR_Y1	**KRG_gaussian**	['RSM_2', 'RSM_4', '**KRG_gaussian**']	1
	HRR_Y2	**KRG_gaussian**	['RSM_2', 'RSM_3', '**KRG_gaussian**']	1

Problem		Promising Surrogate Models		Accurate prediction
		Best Type	Candidate Surrogate_Optional Types (CSOTs)	
	HRR_Y3	**KRG_power**	['RSM_2', 'RSM_4', 'KRG_gaussian']	0
	Blowpipe_Y1	**RSM_6**	['RSM_4', 'RSM_5', '**RSM_6**']	1
	Blowpipe_Y2	**KRG_power**	['RBF_multiquadric', 'RSM_6', '**KRG_power**']	1
	Blowpipe_Y3	**KRG_power**	['RBF_multiquadric', 'RSM_6', '**KRG_power**']	1
Theoretical Problems	Perm500	**MARS_3**	['RBF_multiquadric', 'RSM_6', 'KRG_power']	0
	Perm40	**RSM_3**	['**RSM_3**', 'RSM_4', 'RSM_5']	1
	Perm280	**RSM_6**	['RBF_cubic', '**RSM_6**', 'KRG_gaussian']	1
	Perm160	**RSM_4**	['RSM_3', '**RSM_4**', 'RSM_5']	1
	Perm1000	**MARS_2**	['RBF_multiquadric', 'RBF_cubic', 'KRG_power']	0
	DixonPrice80	**KRG_gaussian**	['RSM_4', 'RSM_6', '**KRG_gaussian**']	1
	DixonPrice500	**RSM_4**	['**RSM_4**', 'RSM_5', 'RSM_9']	1
	DixonPrice400	**RSM_4**	['**RSM_4**', 'RSM_5', 'RSM_7']	1
	DixonPrice300	**RSM_4**	['**RSM_4**', 'RSM_5', 'RSM_8']	1
	DixonPrice200	**RSM_4**	['**RSM_4**', 'RSM_5', 'RSM_6']	1

Problem	Promising Surrogate Models		Accurate prediction
	Best Type	Candidate Surrogate_Optional Types (CSOTs)	
DixonPrice20	**RBF_multiquadric**	['**RBF_multiquadric**', 'RBF_cubic', 'KRG_gaussian']	1
Beala90	**MARS_3**	['RBF_cubic', 'MARS_2', '**MARS_3**']	1
Beala70	**MARS_2**	['RBF_multiquadric', 'RBF_cubic', 'KRG_gaussian']	0
Beala50	**RBF_cubic**	['RBF_multiquadric', 'MARS_3', 'MARS_4']	0
Beala30	**RBF_cubic**	['RBF_multiquadric', '**RBF_cubic**', 'KRG_gaussian']	1
Beala120	**RSM_3**	['RBF_cubic', 'MARS_2', '**RSM_3**']	1
Beala10	**RBF_inverse**	['**RBF_inverse**', 'MARS_2', 'KRG_power']	1

To further validate the performance of the proposed AutoSM, the results are compared with comparable studies to confirm their consistency, as shown in Table 7.8. We bold all consistent results, and it can be found that their results are highly consistent. For the Hot Rod Rolling problem, the predicted surrogate-optional type yielded by AutoSM matches the ‘Best Type’ well.

Table 7.8. Comparisons between our work and other studies in surrogate model prediction

Problems	Other studies	Our work
Perm	RBF_linear, **RBF_multiquadric** (Mehmani et al., 2018) **KRG, RBF** [4]	**RBF_multiquadric**, RSM_6, **KRG_power**
Hartmann-3	**RBF_multiquadric**, KRG_Gaussian (Mehmani et al., 2018), **MARS** (Wang et al., 2014)	**MAR_4**, KRG_power, **RBF_multiquadric**
Dixon and Price	RBF (Liu et al., 2016)	RSM_4, RSM_6, KRG_gaussian
Beala	**RBF, MARS** (Jin et al., 2001)	**RBF_multiquadric**, RSM_6, KRG_power

From Figure 7.8, almost all predicted accuracy of HRR is higher than 80%. Even for the mispredicted types in the HRR_Y3 problem, the accuracy of surrogate-optional type: KRG_gaussian is more significant than 0.98. It indicates the predicted type can achieve excellent performance, even though it is not on the recommendation list. In the Blowpipe design problem, the predicted surrogate-optional types perfectly match the *'Best Type'*. From Figure 7.9, all estimated surrogate optional types have high accuracy in the Blowpipe_Y1 problem, and the same surrogate-optional type (KRG-power) provides all the best solutions for both Blowpipe_Y2 and Blowpipe_Y3 problems.

From the pie chart Figure 7.10, there are many choices for the best surrogate type, and their frequency does not differ much. Based on this finding, we further emphasize that there is no universally dominant surrogate-optional type, which is suitable for all different situations.

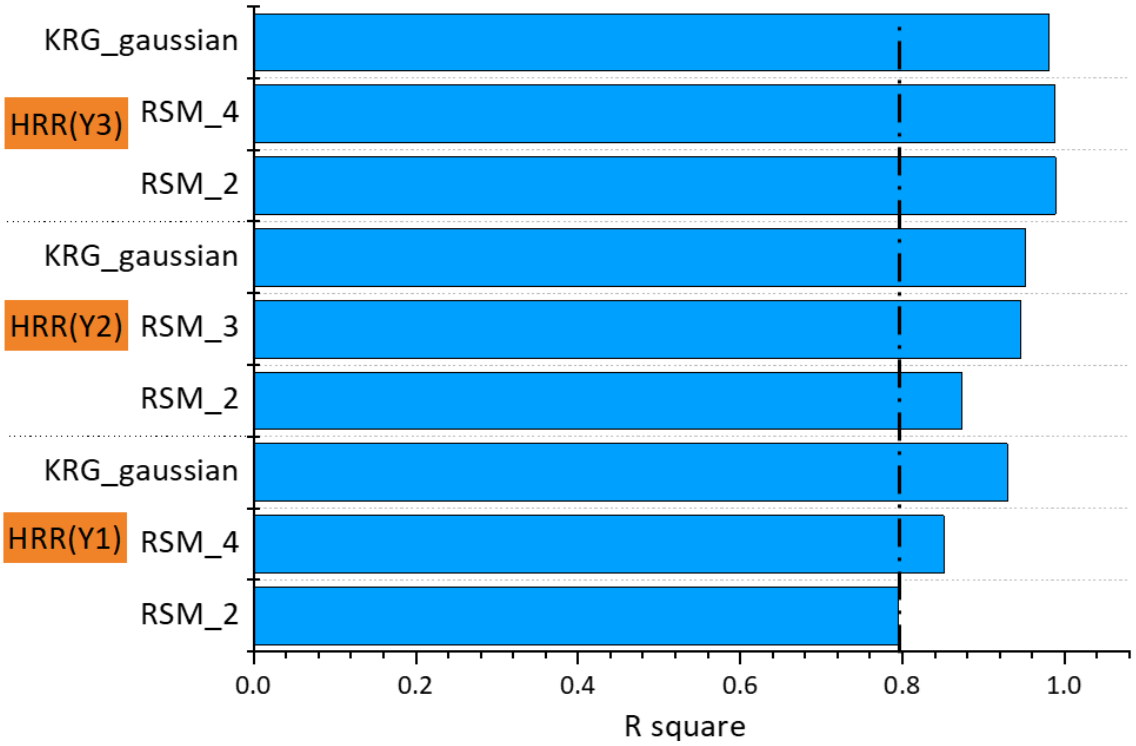


Figure 7.8. Predicted accuracy of hot rod rolling problem

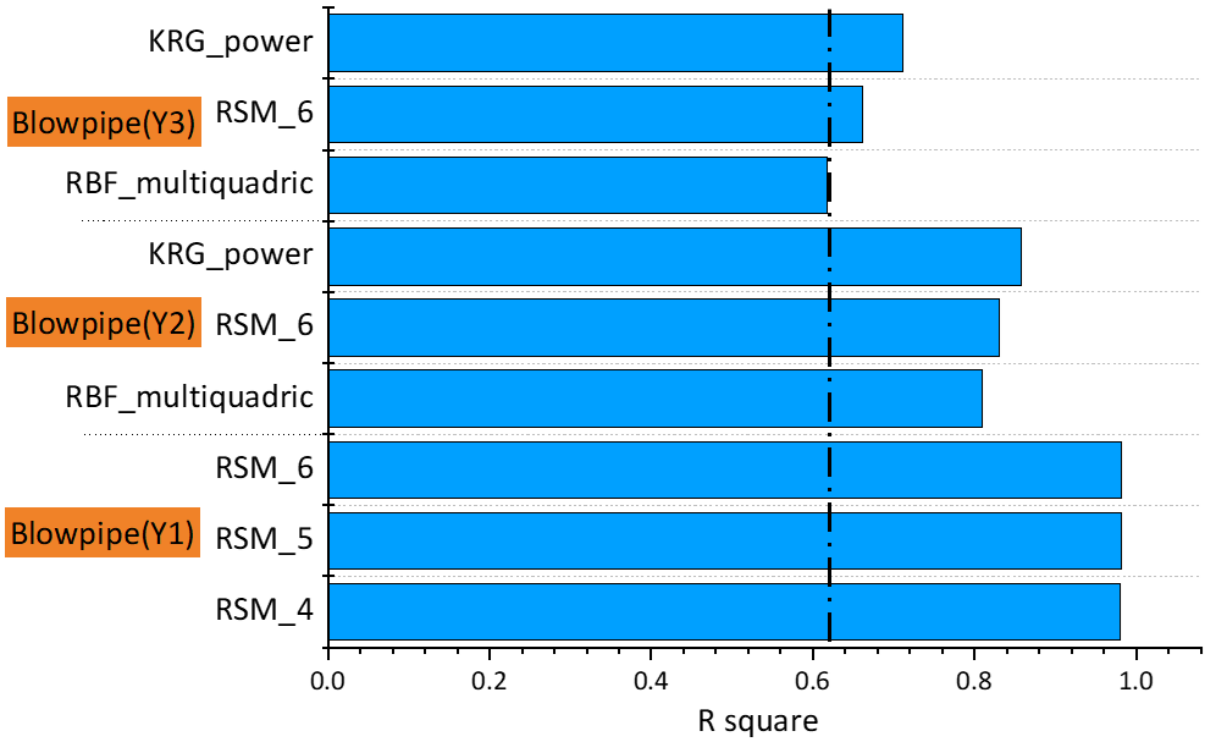


Figure 7.9. Predicted accuracy of the blowpipe design problem

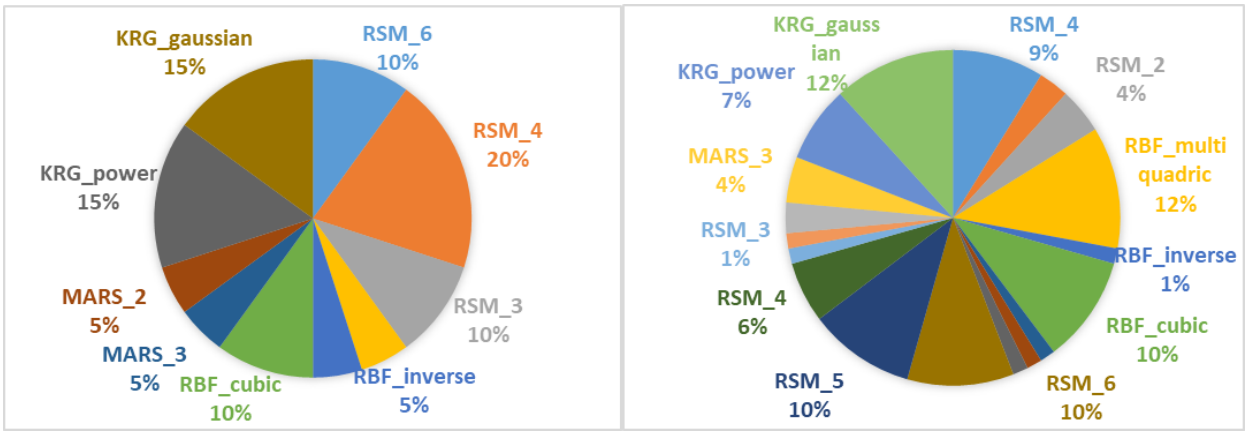


Figure 7.10. Frequency of best surrogate-optional types (Left figure) and predicted surrogate-optional types from AutoSM across 5 test problems (Right figure)

Based on the ten benchmark functions and 5 test problems, we find that there is a negative correlation between '*Nonlinearity*' and '*Predicted Accuracy*'. As shown in Figure 7.11, the fitted curve (solid black line), which is generated to reveal the relation between nonlinearity and accuracy, has a monotonically decreasing trend. To ensure the prediction of this decreasing trend, we give a 95% confidence band in the gray shade for each predicted point, which is generated to reveal the relation between nonlinearity and accuracy. Based on the fitted curve in Figure 7.11, we also show that there is a negative correlation between '*Nonlinearity*' and '*Predicted Accuracy*'. More specifically, when the nonlinearity is low (≤ 0.5), the model has fewer judgment errors $\frac{55}{167} \approx 33\,\%$; however, as the nonlinearity is high (> 0.5), the frequency of judgment errors significantly increases to $\frac{26}{56} \approx 54\%$. The program run time is the computational time required by Python codes in a Lenovo computer i7-4720HQ 8G 128G SSD+1T GTX960M. Three test functions are used for the comparison of the surrogate model selection between our proposed AutoSM and the COSMOS method proposed by (Mehmani et al., 2018). As shown in Table 7.109, the running time of our suggested surrogate model type prediction is about nine times less than the COSMOS method.

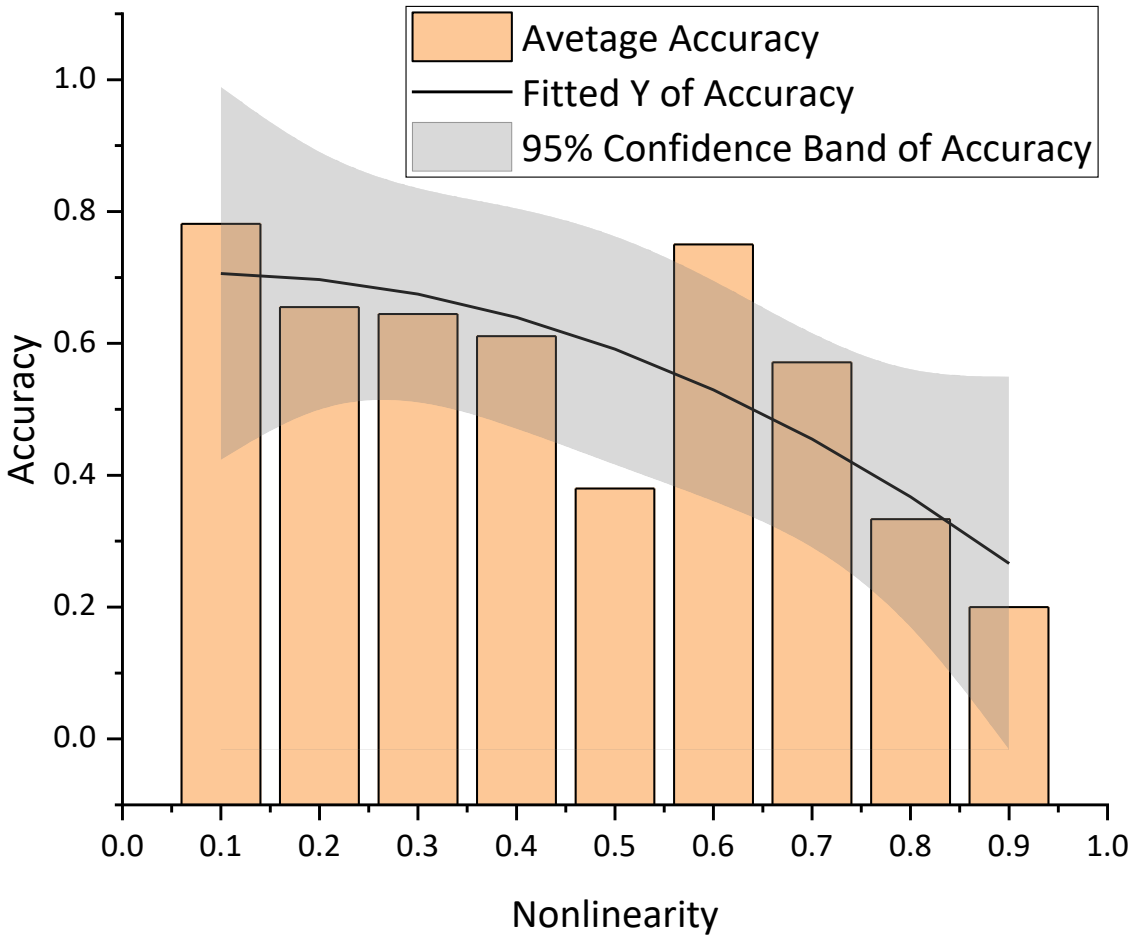


Figure 7.11. A statistical histogram of the relation between nonlinearity and accuracy.

As shown in Table 7.109, the running time of our suggested surrogate model type prediction is about nine times less than the COSMOS method. Concerning the predicted surrogate model type in Table 7.10, the average predicted accuracy of AutoSM of all applications and theoretical problems is 78%, which is way above the average. More specifically, the AutoSM has high accuracy in predicting low-nonlinearity problems (>90%). Overall, the proposed AutoSM can significantly reduce the time for the surrogate model selection while maintaining good prediction accuracy.

Table 7.9. The surrogate model type prediction running time of COSMOS and AutoSM.

Function Name	Running time		
	COSMOS (Second)	**AutoSM** (Second)	**COSMOS/AutoSM** (Degree)
Perm1000	96421s	10120s	9.53
DixonPrice80	12370s	1720s	7.19
Beala120	10230s	1427s	7.17
Average time	39673.67s	4422.3 s	8.97

We believe that the robustness of each surrogate model represents the intrinsic properties of the model itself, and it is independent of the application scenario and test functions. As such, we just use the result of 10 benchmark functions to identify the robustness of each surrogate-optional type. As shown in Table 7.10, we find that KRG has the largest robustness value, which represents that it is susceptible to different problems. The average robustness of RSM and RBF are close and very small, except for the linear type of RBF and the order 2 type of RSM.

Table 7.10. Robustness of surrogate model with surrogate types and surrogate optional types

Surrogate Model	Robustness	
	Surrogate_Optional Type	Surrogate Type
RSM_6	0.1048	
RSM_5	0.0940	
RSM_4	0.0881	0.1183
RSM_3	0.0827	
RSM_2	0.2220	
RBF_thin_plate	0.0845	0.1239
RBF_multiquadric	0.1075	

	Robustness	
Surrogate Model	Surrogate_Optional Type	Surrogate Type
RBF_linear	0.2075	
RBF_inverse	0.1414	
RBF_cubic	0.0788	
MARS_4	0.2596	
MARS_3	0.3565	0.3972
MARS_2	0.5753	
KRG_spherical	0.5174	
KRG_power	0.2283	
KRG_linear	0.3876	0.5790
KRG_gaussian	0.1648	
KRG_exponential	1.5971	

As shown in Figure 7.12, we further illustrate the combined performance of each surrogate-optional type in terms of average accuracy and robustness. In the model selection work, there are many trade-offs between accuracy and robustness. However, some surrogate optional types maintain a high degree of robustness and accuracy, for example, RBF-cubic type, RBF-multiquadric type, and RSM-4 type. The significance of this work is that even in the absence of prior knowledge, AutoSM can be used to recommend some surrogate models with good accuracy and robustness. Based on the training data generated from 10 benchmark functions, a simplified tree structure is illustrated in Figure 7.13.

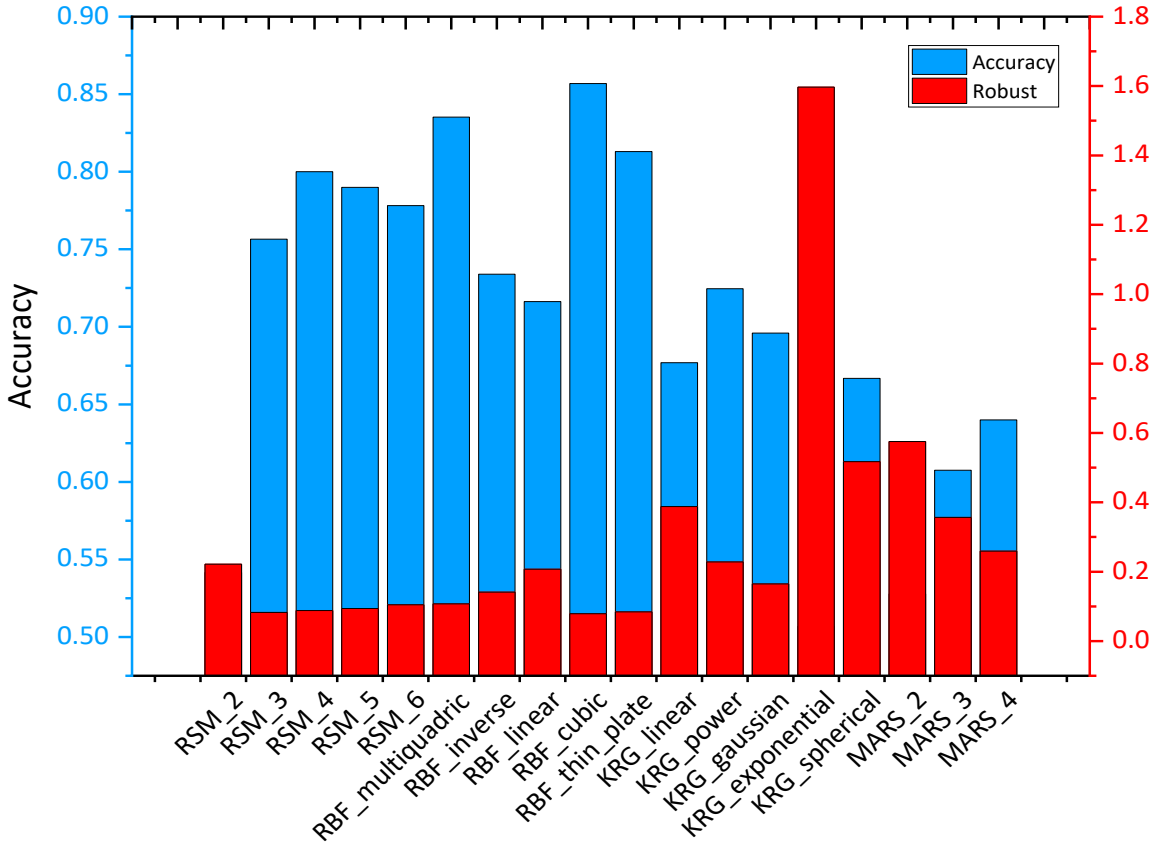


Figure 7.12. Average of accuracy and robustness of each surrogate-optional type

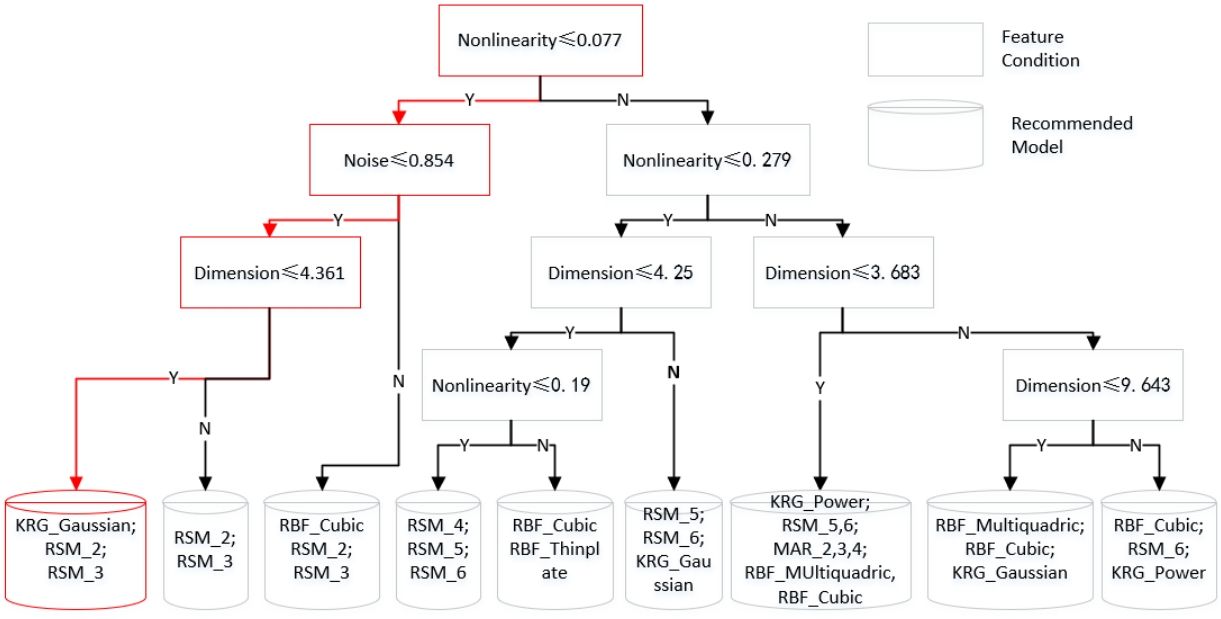


Figure 7.13. Categorization of surrogate-optional types based on accuracy for multi-label CART

Although this tree structure is not a complete classification of the surrogate-optional types of each category, the number of selected type candidates has already been reduced from 18 to 2 or 3. from Figure 7.13., the extracted selection rules are the combination of the intervals of these four features instead of their specific values. For example, as shown by the marked part in this figure, the selection rule is shown as follows:

***'IF* $\mathbf{Nonlinearity \leq 0.077}$ & $\mathbf{Noise \leq 0.854}$ & $\mathbf{Dimension \leq 4.361}$:**

THEN KRG_gaussian, RSM_2, RSM_3'.

These rules exist in the well-trained selector CART, and once we use the selector, it automatically outputs the recommend surrogate-model types. For the number of rules, it is depended on the training process and the depth setting of CART. In our work, we set the max depth to 6, and the number of rules is more than 50. Besides, generating multiple types of output is beneficial to improve the robustness of our AutoSM. With this interpretable illustration of proposed AutoSM, serialization rules of surrogate model selection can be easily obtained.

7.6 On Verification and Validation Creating an Automatic Multi-Level Surrogate Model Selection Process

7.6.1. Theoretical Structural Validation

Theoretical structural validation refers to accepting the validity of automatic surrogate model selection and accepting the internal consistency of the way the putting together all three steps of surrogate model selection, including surrogate type, function type and hyperparameter tuning. Theoretical structural validation involves systematically identifying the scope of the automatic surrogate model selection, reviewing relevant literature and identifying the research gaps that is existing, identifying the strengths and limitations of the surrogate model selection process used based on literature review, determining the gaps and approaches that can be leveraged for SM

selection, reviewing literature on the advantages, disadvantages and accepted domains of application, and checking the internal consistency of the constructs both individually and when integrated.

In Chapter 3, surrogate modeling literature is reviewed in detail. A detailed comparison of manual and automated SM selection is then carried out in Section 7.2.1. Then, the evolutionary algorithm based surrogate model selection is critically evaluated in Section 7.2.2. Then, the proposed three-layer structure of surrogate model selection is discussed in Section 7.3.1 followed by the introduction of AUTOSM framework in Section 7.4. Gap 1 in this dissertation, which is on Classifying the surrogate models based on common selection criteria. To address this gap, we framed the Research Question 1, which is "what are the main classes of the design of experiment (DOE) methods, surrogate modelling methods and model-fitting methods? ". To address this research question, we hypothesized that surrogate modeling methods can be classified based on the problem size, computational time, and accuracy. By doing this project and completing Chapter 3 of the dissertation, we achieved the key outcome of a framework to provide guidance for researchers and practitioners to choose the most appropriate surrogate model based on incomplete information about an engineering design problem.

7.7 Closing Remarks on Creating an Automatic Multi-Level Surrogate Model Selection Process

The implementation of surrogate modeling techniques has become a substantive practice in engineering due to the expensive computation of current simulation codes, and how to select the appropriate surrogate model becomes a critical problem blocking the entire model constructing process. In this work, we propose a knowledge involved selection method, called AutoSM, to tackle the surrogate model selection problem. Compared to previous model selection approaches,

our AutoSM has two primary advantages: i) developing an interpretable decision tree model (CART) to generate the selection knowledge and reduce the number of surrogate model candidates, which increase the surrogate model selection speed dramatically; ii) having more comprehensive considerations of engineering scenarios by involving more complete features of engineering problems.

We mapped the difference of problems into the difference of datasets using a set of features. So, we summarize four typical features for ten benchmark functions. By using instantiations as training data, CART is implemented to build the relationship between problem features and appropriate surrogate-optional types. In this way, the initial candidate model types are reduced from a ‘pool’ to a ‘cup’, and the selected speed is significantly improved.

Three theoretical problems and two practical design problems, including a Hot Rod Rolling Process problem and a Blowpipe design problem, are employed to test the performance of AutoSM. Based on the results, using the proposed AutoSM method, we decrease the running time for each benchmark function by about 90% while maintaining the level of accuracy and robustness, which is a huge improvement. We also illustrate that the predicted accuracy of AutoSM is reached nearly 78% for all test problems. For the Blowpipe design problem, the anticipated results of AutoSM have an accuracy approaching 100%. Besides, RBF-cubic, RBF-multiquadric, and RSM-4 show good prediction accuracy and robustness on almost all of the test problems. As such, these three surrogate models can be the first choice for surrogate models used in engineering design. Finally, because of the interpretable ability of CART, a visualized selection rule tree is created to decrease the required time on training and to select models on the evolutionary algorithm-based model selection process. Using this rule-based tree, we also provide an idea for solving the model selection problem rapidly.

CHAPTER 8 DESIGNING EVOLVING CPS SYSTEMS TRANSITIONING FROM PUSH TO PULL LOW CARBON ECONOMY

(Comprehensive Problem)

In this chapter, a comprehensive problem is formulated to demonstrate the utility of the partitioning-approximation-coordination design architecture. The Hypothesis 1 is tested in this chapter. An evolving cyber-physical-social system transitioning from a push to a low carbon pull economy is modeled to verify and validate the proposed partitioning-approximation-coordination design architecture. A multi-echelon, multi-channel, multi-commodity supply chain (SC) problem is modeled and designed from a climate change mitigation perspective. Research questions 1, 2, and 4 are addressed in this chapter and Hypothesis 1 is validated.

8.1 Frame of Reference on Green Supply Chain Design

Global warming, resulting from the use of fossil fuels, threatens the environment. Energy efficiency is one of the most important ways to reduce this threat (Alizadeh et al., 2016a; Alizadeh et al., 2016c; Alizadeh et al., 2015a). Moreover, increasing demand for energy, coupled with a restricted supply in world markets, results in an increased price for energy. This general development, as well as dynamics in price setting, generates uncertainties for organization schemes with respect to accurately calculated energy costs (Alizadeh et al., 2016b). The situation becomes even more complex since proper improvements in energy efficiency may be achieved by an approach that considers multiple activities and the energy consumption system. Fossil fuel combustion causes global warming by releasing CO_2 and other greenhouse gases (GHGs) to the atmosphere (IPCC and Team, 2014). As shown in Figure 8.1, transportation accounted for 29% and electricity generation accounted for 28% of GHG emissions in the United States in 2019 [7].

Among transportation based GHG emissions, the gasoline consumption by personal vehicle owners has the largest portion, at 81.9%.

Transportation and electricity use of a retailer's supply chain network (like Walmart) are the main factors in determining its performance, which makes the supply chain network an interesting case in terms of GHG emission reduction (Jin et al., 2014). The fuel efficiency of the vehicles, the amount of load that they carry, the traveling distance, the capacity of the facilities primarily controls the GHG emission in a supply chain network (Benjaafar et al., 2012).

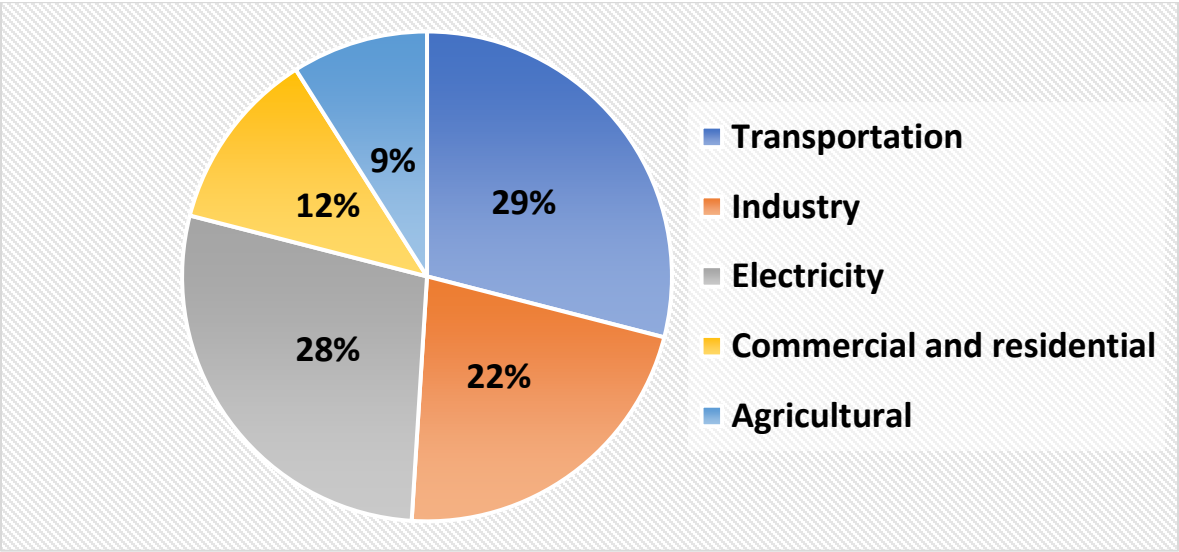


Figure 8.1. Total emissions in 2019 = 6,457 million metric tons of CO_2 equivalent (data source is (EPA, 2019) and figure is drawn by authors).

The literature on green supply chain (GSC) network designs covers a broad range of topics, including facility location problems, location assignments, and vehicle routing problems. The goal of the supply chain (SC) network design problem is to find less operation cost (including variable production cost, transportation, and constant operating cost of facilities) (Cachon, 2014). Nevertheless, the goal of GSC network design is to identify the facility locations, facilities, and distributing resources to achieve less total cost of operation and GHG emissions (Altmann, 2015;

Brandenburg et al., 2014; Kuo et al., 2018; Osmani and Zhang, 2014; Seuring and Müller, 2008; Shaw et al., 2016). Overall GHG emission of the SC is identified by considering *transportation* and *storage* processes.

Transportation-based GHG emissions are calculated from the fuel usage of vehicles multiplied by a conversion factor, and *storage*-based emission caused at facilities is calculated from the electricity usage of facilities multiplied by a conversion factor (Pishvaee et al., 2012; Soysal et al., 2014; Zakeri et al., 2015). Thus, a conventional SC network design based on only operating cost is restructured into a network design based on GHG emissions. Various proposals to decrease GHG emissions (for example, carbon tax and carbon credits) have been combined with traditional SC constraints (for example, demand, capacity, supply, and storage balance constraints). Likewise, the objective function has also been changed into a combination of operation and emission costs (Choudhary et al., 2015; Colicchia et al., 2016; Garg et al., 2015).

In addition, (Nouira et al., 2016) studied the effect of emission-sensitive demand over the facility location problem. The emission reduction problem has also been combined into a closed-loop SC network design and reverse logistics (Das and Rao Posinasetti, 2015; Gao and Ryan, 2014; Garg et al., 2015; Kannan et al., 2012; Mohajeri and Fallah, 2016; Tao et al., 2015). A green SC network design is a decision-making problem which is mostly modeled by a mixed-integer linear programming method.

One class of vehicle routing problems that has been extensively studied is the Traveling Salesman Problem (TSP) Assuming a network of *n* stores, the TSP is to find the shortest path between the *n* points, starting and ending at the same store. Many studies have been conducted on heuristics and algorithms for the TSP (Applegate et al., 2003; Bektas, 2006; Bertsimas, 1992; Bock and Klamroth, 2019; Braekers et al., 2016; Bramel and Simchi-Levi, 1997; Carter and Ragsdale,

2006; Chatterjee et al., 1996; De Koster et al., 2007; Durbin and Willshaw, 1987; Fischetti et al., 1997; Helsgaun, 2000; Jaillet, 1988; Laporte, 1992; Laporte and Osman, 1995; Lawler, 1985; Padberg and Rinaldi, 1991; Potvin, 1996). The TSP is used in some SC designs for routing the trucks, starting at a warehouse, visiting each store, and returning to the warehouse.

Other vehicle routing problems have been studied, focusing on how a group of vehicles should be used to deliver to a group of points in a region in order to minimize the total distance travelled (for example, Daganzo (Daganzo, 1984), Dantzig and Ramser (Dantzig and Ramser, 1959), Haimovich and Rinnooy Kan (Haimovich and Rinnooy Kan, 1985) and (Bektaş et al., 2016; Bertazzi and Secomandi, 2018; Braekers et al., 2016; Campelo et al., 2019; Chiang et al., 2019; Defryn and Sörensen, 2017; Erdoğan, 2017; Govindan et al., 2019; Kalayci and Kaya, 2016; Koç et al., 2016; Li et al., 2019; Neves-Moreira et al., 2019; Niu et al., 2018; Reyes et al., 2017; Ruiz et al., 2019; Salavati-Khoshghalb et al., 2019; Soleimani et al., 2018; Uchoa et al., 2017; Vincent et al., 2017; Wei et al., 2018; Xiao and Konak, 2017)).

These studies are further expanded by research which consists of inventory control and vehicle route planning (for example, Gallego and Simchi-Levi (Gallego and Simchi-Levi, 1990), Burns and co-authors (Barnes and Langworthy, 2003), Federgruen and Zipkin (Federgruen and Zipkin, 1984) and (Bouma and Teunter, 2016; Chitsaz et al., 2016; Crama et al., 2018; Fokkema et al., 2020; Jafarian et al., 2019; Mirzapour Al-e-hashem et al., 2017; Soysal et al., 2019)). The main distinctions between TSP and this GSC design are (i) the retailer can select the stores' locations and (ii) the retailer is responsible for the customers' traveling expenses (for example, the "last mile" of the SC is not neglected (Jiang et al., 2019; Kitjacharoenchai et al., 2019; Wang, 2019; Zhou et al., 2018a)).

Concentrating on just the customers' traveling costs, the problem is equivalent to the famous k-median problem (also called p-median, the location-allocation problem and the multi-source Weber problem). In general, the goal is to partition a network (typically a complete graph of all points in the network) into *k* partitions, such that the distance between each point and its partition center is minimized (Figure 2). In Figure 2, circles are nodes, and squares are the partition centers. Line segments indicate the distance between a node and its partition center, and partitions are colored. In the context of SC's, the goal is to identify *k* locations as the stores to minimize the overall traveling charge from the demand locations (customers) to the closest store.

The k-median problem is usually investigated in its discrete form (for example, a limited number of potential demand and store locations (Kuzmenko and Uryasev, 2019)). However, there are studies on the continuous k-median problem, which is the SC design problem where the retailer's traveling and space charges are neglected (Xavier and Xavier, 2019). Research on the k-median problem has concentrated on good solution processes instead of building the solution. For more detailed information on the k-median problem refer to Daskin (Daskin, 2011). Extensive solution algorithms have been studied by Brimberg and co-authors (Brimberg et al., 2000) for the discrete k-median problem, and Fekete and co-authors (Fekete et al., 2005) studied the continuous form of the problem.

Since the k-median problem creates clusters, it is used to solve the facility location problem. In previous literature, a dual approach of the k-median problem and TSP to determine facility locations has been considered. Cachon (Cachon, 2014) studied this integrated problem, modeling vehicle routing as a k-median problem and truck routing as a TSP.



Figure 8.2. A network partitioned using the k-median algorithm.

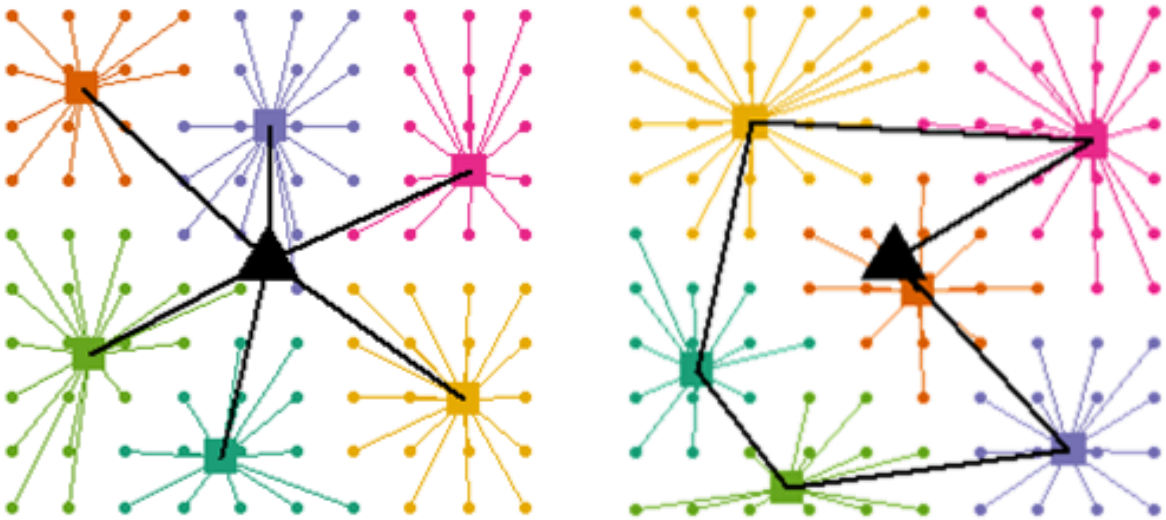


Figure 8.3. A two-echelon k-median network (left), and a two-echelon k-median and TSP network (right).

We consider a two-echelon network, consisting of two instances of the k-median problem: one to model customers driving from their homes to a store, and another to model trucks driving from a warehouse to a store. In both cases, the problems are solved simultaneously—that is, the cars and trucks are routed simultaneously, and determine the location of the stores. In Figure 8.3, the difference between our model (left), and Cachon's (right) is shown. The triangle in Figure 8.3. is

the second level partition center. For a SC, circles are customers, squares are stores, and the triangle is a warehouse.

In a GSC network design, the goal is to find the SC network architecture and the capacity of the facilities in a way which the overall SC cost and GHG emission is low at the same time (Altmann, 2015; Purohit et al., 2016; Qi et al., 2017). There is extensive research on the relationship between GHG emissions and operational decisions. Haddadsisakht and Ryan study a SC network design in which the company can choose from a group of transportation modes under uncertain carbon tax and stochastic demands that change in their replenishment lead time, amount of emission and delivery charge (Haddadsisakht and Ryan, 2018).

Benjaafar and co-authors (Benjaafar et al., 2012) investigate a single-location model where inventory planning decisions affect SC inventory charges, GHG emissions, and back-order charges which comprise an average constant quantity per unit kept in inventory; a constant quantity per unit sold; and a constant quantity per delivery. In our model, store locations are not pre-determined and will be located as part of the solution. Also, dissimilar to Haddadsisakht and Ryan (Haddadsisakht and Ryan, 2018), the retailer in our model has a single mode of transportation and the main focus is on distances travelled rather than lead time. Unlike (Benjaafar et al., 2012), our proposed model does not contain constant GHG emissions per unit sold, and the GHG emissions per delivery is not constant (it depends on the number and location of the stores).

Similar to (Haddadsisakht and Ryan, 2018), in our proposed model, incurring an explicit cost for GHG emissions is not likely to considerably cut GHG emissions unless the cost is unreasonably high. In a similar way, Chen and co-authors (Chen et al., 2013) study a model where a retailer and customers charge traveling and GHG emission charges based on distances. They also assess the charge minimizing the number of stores. However, they use a dissimilar spatial geometry, in which

stores, and customers are placed on the border of a circle. They find that restricting a retailer's GHG emissions might result in an increase in overall GHG emissions, since it may result in higher customer GHG emissions. These sections show the effectiveness of various methods of providing incentives to cut GHG emissions (for instance via carbon taxes or different restrictions). We conduct a sensitivity analysis to further investigate this issue.

Some authors study a SC in which companies can invest in GHG emissions reduction (Zhou et al., 2018c). They consider the amount of inventory, store locations, and replenishment paths. That is, they investigate how various models for allotting GHG emissions to different operations affect the quantity of investment in GHG emission cuts. Some scholars also investigate carbon tax regulations and find that poorly selected regulations may lead self-interested companies to choices which increase total GHG emissions (Jakhar, 2015; Keskin and Plambeck, 2011; Kök et al., 2018; Plambeck, 2012). Zhu and co-authors (Zhu et al., 2016) analyze the upstream architecture of the SC, keeping the downstream architecture, the location, and the number of stores constant. There is extensive literature on facility location problems (see Daskin 1995 and Snyder and Shen 2011) in which the authors usually concentrate on the constant charge of opening facilities and the transportation charge of serving the customers from facilities which are open (Daskin, 2011; Snyder and Shen, 2011). Govindan and co-authors (Govindan et al., 2017) review extensive literature which comprises inventory charges in the facility location problem: for instance, Correia and Melo (Correia and Melo, 2016), Cui and co-authors (Cui et al., 2010), and Klose and Drexl (Klose and Drexl, 2005). A critical observation from these studies is that there is a trade-off between economies of scale in inventory and location, where fewer and larger warehouses are more inventory effective. However, that method raises transportation charges as the warehouses are more distant from consumers. This is the same famous trade-off concerning economies of scale

and inventory. This trade-off also exists in our model with retail stores. Therefore, if there are more stores, they are less inventory efficient since they meet less demand. They also have a single level of transportation in the SC (between warehouses and stores) while this model reflects the interaction between two transportation stages of the SC (inbound to stores and to customers' houses). Most importantly, they do not concentrate on the ramifications of failing to reflect crucial charges inside the SC (as in the impact of neglecting GHG emissions charges).

Among these distinctions, assumptions regarding the mode of transportation are a critical issue in making the problem more realistic. For example, in many studies, including Cachon (Cachon, 2014), the assumptions are: (1) the supplier trucks use the traveling salesman problem, which is not always appropriate; (2) a specific predefined distribution of demand such as uniform or normal distributions for the demand (Cachon, 2014). So, as shown in Table 1, two methodological contributions of ours are to remove these two assumptions. The first contribution is to remove the reliance on the TSP, as used in previous studies including Cachon (Cachon, 2014). Instead, we consider a SC based on the k-median problem. This is more appropriate for large retailers, such as Walmart, that utilize high-load truck resupply strategies and cross-docking. Moreover, while the total path distance (that is, the distance to go from the warehouse to every store and back) is minimized in the TSP, this does not make for a desirable supply chain design—consider the case in which the first store in the tour requires three truckloads of goods. Then, it would be optimal to send the three trucks, then have them return directly to the warehouse, instead of visiting every other store. However, the TSP would have them do the later. As such, we consider the vehicle routing problem to be a k-median problem, where the supplier truck starts from the warehouse, delivering the goods to the stores based on their capacity and demands, and then comes back to the warehouse.

Additionally, some authors assume that the supply chain is a hierarchical multi-echelon system (Altmann, 2015; Osmani and Zhang, 2014) while we considered it as a two-echelon network-based problem. The second contribution is to remove the second assumption in the literature by designing the supply network based on population density data, which has a non-uniform distribution. A green supply chain which has the features highlighted in gray in Table 8.1 is closer to how large retailers like Walmart operate. In this section, we focus on the downstream supply chain, which consists of inbound replenishments to facilities, the facilities themselves, and the portion between the retail facilities and customers' homes. As the number of facilities grows, a dense network is created, facilities become smaller, and customers find themselves closer to a facility, so they do not need to travel long distances for shopping.

The capacity of retail facilities is of interest to city planners. One of the challenges is that large retail facilities result in a car culture, which makes customers travel longer distances than what they would travel if smaller facilities existed nearer their homes (Fry and Owen, 2013; Glaeser, 2013; Shoup, 2017). Since traveling distance by customers results in additional emissions, environmental experts have been suggesting larger facilities (Cachon, 2014). The challenge of retail facility capacity is also related to a retailer's long-term planning. First, a retailer's store size impacts its attractiveness to customers.

If we assume all other features the same, a customer prefers the convenience of a close store rather than a far store. For instance, (Pancras et al., 2012) finds customers act in such a way that the cost of one traveling mile is $0.60 for a fast food supply chain. Second, the problem of capacity is also vital to suppliers with emission reduction goals. For instance, Walmart has promised that it mitigates one billion m^3 tons of GHG emissions from its SC network by 2030 (Walmart, 2018b).

These drops can happen from the retailer network it directly manages (for example, falls in its fuel use) that are often called scope 1 GHG emissions.

Table 8.1. Summary of the critical evaluation of the literature on GSC design problem.

	Inventory management		Truck Routing		Design		Demand Distribution	
	Size	Location	TSP	k-Median	Hierarchy	Network	Deterministic	Random
(Pishvaee et al., 2012)	*	*	*			*	*	
(Osmani and Zhang, 2014)	*	*	*		*			*
(Altmann, 2015)	*	*	*		*		*	
(Shaw et al., 2016)	*	*	*			*		*
(Kuo et al., 2018)	*		*			*	*	
(Soysal et al., 2014)		*	*			*		*
(Zakeri et al., 2015)	*		*		*			*
(Choudhary et al., 2015)	*	*	*			*	*	
(Garg et al., 2015)	*		*			*	*	
(Colicchia et al., 2016)		*	*			*	*	
(Kannan et al., 2012)		*	*			*	*	
(Mohajeri and Fallah, 2016)		*	*		*			*
(Gao and Ryan, 2014)		*	*			*		*
(Garg et al., 2015)	*	*	*			*	*	

	Inventory management		Truck Routing		Design		Demand Distribution	
(Cachon, 2014)	*	*	*			*	*	
Our work	*	*		*		*		*

On the other hand, decreases can descend further into the retailer network (for instance, decreases in its customer fuel use), which is a share of scope 3 GHG emissions. Even though it is possibly too expensive to change the entire store's capacity to obtain emission mitigation goals in the short-run, the capacity and position of facilities is considerably changed through a long-run period of five or more years. For instance, from 2006 to 2010, Walmart reduced the number of discount facilities in the US by one-third (1,209 to 803), and increased bigger supercenters by 60% (1,713 to 2,747) whereas also growing its small store setup, Neighborhood Marketplaces, by about 60% (100 to 158) (Cachon, 2014).

The remainder of the section is as follows: in Section 8.3, we explain the generic modeling process of a green supply chain. In Section 8.4, we introduce and explain our approach that we use for designing the GSC. In Section 8.5, we discuss the results of an extensive sensitivity analysis, and finally, we end with the closing remarks in Section 8.6.

8.2 Modeling the Green Supply Chain Problem as a Two-Echelon Supply Chain

8.2.1. Modeling decision interactions

In this section, we explain how the retail store density problem is related to the downstream supply chain. The requirements for solving the problem include the number, location, size of the suppliers and stores, the number and schedule of the deliveries, and the shape of the area. Also,

research questions worthy of investigating and hypothesis to the test these research questions is discussed in this section.

The retail store density problem is stated as: "given a set of customers spread non-uniformly across some region, determine how many stores should a retailer have in the region, and where should those stores be, in order to minimize the store's operating costs."

Our SC is modeled as a two-echelon network, in which goods flow from a single, central warehouse to stores, then from the stores to customers. Trucks transport the goods from the warehouse to the stores, and cars from the stores to the customers. Neither locations nor the number of facilities are known ahead of time. Both vehicle routing problems (cars and trucks) are modeled as k-median problems.

In our SC, a retailer's operating cost consists of three components: stores, trucks, and cars. We assume that while demand is non-uniform across a region, it is uniform across customers, that is, while customers may be grouped in various neighborhoods, each customer purchases the same amount of goods from a store over a given period. Of course, if a retailer only needs to minimize the cost of operating their stores, the retailer should simply open only one store. However, embedded in the store's operating cost is the customer's cost: because customers are willing to pay more to enjoy the convenience of going to a closer store, retailers compensate consumers for driving long distances by having lower prices at larger stores (Grewal et al., 2012).

Small stores in densely populated areas charge more because consumers are willing to pay higher prices for the convenience of a closer store. Meanwhile, large stores charge less, compensating customer for the longer distance they drive to get to the store. As such, the retailer has costs: (1) the truck cost, which is the cost of a truck driving from a warehouse to the store; (2) the car cost, which is the cost of a car driving from the customer's home to the store and the retailer

reimburses the customer for; (3) the inventory cost, which is the cost of having a store large enough to fulfil all of the customer's needs.

We use a single warehouse in our model, which can be extended to a multi-warehouse model. In our model, each store has a truck that connects directly to it from the warehouse. Each period, the store is resupplied by this truck, perhaps many times (or by many trucks at once – the two are equivalent). We assume that a retailer can stockpile goods, and thus if a store does not need enough products to fill a truck in full, they can fill a truck in one period, stockpile the goods, then use it from the stockpile in the next period. In this way, we model the truck-resupply problem as a k-median problem, in contrast with previous literature, which modeled the truck-resupply problem as an instance of the Traveling Salesman Problem (Cachon, 2014).

The customer connections are also a k-median problem, which allows us to model the Retail Store Density Problem entirely as a partitioning problem. Given a set of customer locations, we wish to partition the set into a number, k, of subsets such that the operating cost is minimized. k is not known ahead of time and must be determined. We also wish to determine the locations of the facilities that each subset of customers connects to.

The inventory cost is interesting for several reasons. Regarding the retail space, if it is proportional to the quantity of inventory kept, a new inventory model is required. In this situation, the optimum policy of selecting inventory quantity is not known and probably intricate. Heuristic strategies have been proposed (Cachon, 2001; Gürbüz et al., 2007), but their findings do not suggest closed-form approximations of inventory levels. Therefore, we propose two different modeling mechanisms. The first is linear: for each customer that connects to a store, a store needs additional space proportional to the amount of goods that a customer purchases per period. This has the advantage of being very easily modeled, as we see in the next section. However, this is not

very realistic – in general, the inventory cost does not scale linearly with each customer added; the cost of the 1000th customer connecting to a store is less than the cost of the 100th connecting. One approach to deal with this non-linearity is to use piecewise cost functions, as we see in the next section.

Related to the Retail Store Density Problem and the k-median problem is the capacitated Facility Location Problem (FLP). It is stated informally as: "given a set of facilities with costs to open each one, and a set of customers with costs to connect to each facility, determine the subset of facilities to open such that all customers are connected to one open facility." We propose a modification to the FLP, in which the partial cost of an additional truck delivering goods to a facility due to a customer connecting is included in the cost for that customer, meaning that the total customer connect cost is $c_c + dc_t / q_t$ where d is the demand or amount of goods (in kg) that a customer purchases in a given period and q_t is the capacity of the truck.

8.2.2. Problem formulation of green supply chain design as a retail store density problem and a linear program

We formulate the Retail Store Density Problem as a linear program. Let f_j refer to the cost of opening a store j, c_{ij} refer to the cost of a customer i connecting to store j, x_{ij} be a binary variable tracking if customer i connects to store j and y_j be a binary variable tracking if store j is open. Then, the objective function is:

Minimize:

$$\sum_{i,j} c_{ij} x_{ij} + \sum_{j} f_j y_j \qquad \text{Equation 8.1}$$

Subject to:

$$\forall i : \sum_j c_{ij} = 1 \qquad \text{Equation 8.2}$$

$$\forall i, \forall j : y_j - x_{ij} \geq 0 \qquad \text{Equation 8.3}$$

$$\forall i, \forall j : x_{ij} \in \{0,1\} \qquad \text{Equation 8.4}$$

$$\forall j : y_j \in \{0,1\} \qquad \text{Equation 8.5}$$

In Equation 8.2, it is shown that every customer must be connected to exactly one store. Equation 8.3 implies that no customer is connected to a store that is closed. Equation 8.4 and Equation 8.5 constrain x and y to be binary.

This objective function works well when the inventory cost linearly scales with the number of customers connected to a store because the customer connects cost can simply include the partial inventory cost for that customer. However, when the inventory cost does not scale linearly, we add the inventory cost to the open cost for a store (f_j) and add an additional constraint that each store must have fewer customers than some maximum capacity q_f:

$$\forall j : \sum_i x_{ij} < q_f \qquad \text{Equation 8.6}$$

In this way, we introduce a piecewise, non-linear inventory cost. With this constraint, each store has a set size (q_f), and there is some set amount of store sizes and the maximum number of customers that can connect to a store of that size. However, this constraint comes at the cost of

computational complexity: if we wish to add just additional possible store sizes, each candidate store location must be modeled as a collocated candidate location with the additional constraint:

$$\forall l : \sum_{k} x_{l*k} \leq 1 \qquad \text{Equation 8.7}$$

Where l replaces j as the store location index and k represents an index in the range of 1 to the number of locations. For instance, if we wish to offer three possible store sizes for 12 possible store locations, our model treats this as 36 possible store locations, with the constraint that collocated stores cannot be simultaneously open (the above constraint).

We generate the costs for the objective function via the following cost functions. The cost of a customer i connecting to a particular store j is the cost of a customer driving to a store plus the partial cost of the truck driving to the store. The cost of the customer driving to the store is:

$$d_{ij}(v_c + f_c(p_f + e_c p_e)) \qquad \text{Equation 8.8}$$

Where d_{ij} is the distance from the customer to the store, v_c is the non-fuel variable cost for the car to travel per unit of distance ($/km), f_c is the amount of fuel required per unit of distance (L/km), p_f is the price of fuel per unit of fuel ($/L), e_c is the amount of emissions due to the car per unit of fuel (kg-CO2 / L), and p_e is the price of emissions ($/kg-CO2).

The cost of the truck driving to the store from the warehouse per customer, is:

$$q_c d_j \frac{v_t + f_t(p_f + e_t p_e)}{q_t}$$ Equation 8.9

Where d_j is the distance from the warehouse to the store, q_t is the amount of goods (in kg) carried by the truck, and q_c is the amount of goods purchased by each customer. The remainder of the variables are as described above, applied to the truck.

Therefore, c_{ij} (the cost of a customer connecting to a store) is:

$$c_{ij} = d_{ij}(v_c + f_c(p_f + e_c p_e)) + q_c \frac{d_j(v_t + f_t(p_f + e_t p_e)}{q_t}$$ Equation 8.10

where q_c is the amount of goods (in kg), which a customer purchases in a given period (that is, the demand). The cost, f_j, of a store j opening, is the cost of the building plus the cost to heat and cool the building:

$$f_j = c_{max} q_c \frac{v_s + f_s(p_f + e_s p_e)}{q_s}$$ Equation 8.11

Where v_s. is the variable cost of the store per unit of space required (for instance, rent), f_s is the fuel required to heat or cool the store per unit of space, p_f is the price of fuel per unit, e_s is the amount of emissions generated by the store per unit of fuel, p_e is the price of energy, and q_s is the number of units stored per unit of space, c_{max} is the maximum number of customers that can connect to a given store, and q_c is the amount of goods that a customer purchases in a given period.

8.2.3. Solving the designed green supply chain and the retail store density problem

The schematic of the proposed GSC network design model is shown in Figure 8.4 (reproduced from above), with circles representing customers, squares representing stores, and the central triangle representing the warehouse. Line segments show connections, and partitions are colored.

The schematic of the proposed method is shown in Figure 8.4. We model the network of customers and facilities using R. The customer's location comes from population density data: each customer is located at some point on an x-y grid. We later discuss choosing the store locations, for now, assume we have some candidate store locations on an x-y grid. Then, we can calculate the distance between each customer and store to create a maximally dense distance graph, which we store as an array. From this, we can calculate the car and truck costs for each customer connecting to each store.

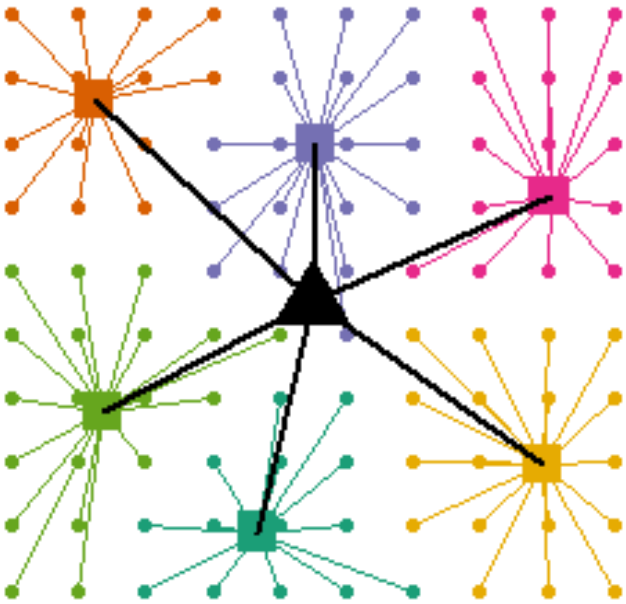


Figure 8.4. The proposed GSC network design model.

Recall that each candidate store location is represented as several candidate facilities, each with a different maximum size and opening cost, with the constraint that no two facilities at a single location are open simultaneously. As such, we have three arrays: (1) *connect*, an array containing

the cost of each customer to connect to the store, for each pair of customer and candidate store location. This is useful to retain as a two-dimension array, because we may wish to index by both the customer and the store. Additionally, it is useful to store the cost of the customer connecting to each candidate store at some location in computer's memory, although their connection cost is the same for each store size. As such, *connect* has size $i * (j * k)$, where i is the number of customers, j is the number of store locations, and k is the number of store sizes. (2) *open* in an array containing the cost to open each store. In practice, this can differ for each candidate store, because there may be different costs (such as the price of electricity or the cost to lease or purchase a store) for each location. We do not model such scenarios and consider the price to open a store of a given size to be uniform. However, we still include the price to open each candidate store in the *open* array for later use. As such, *open* has size $j * k$ (3). The array *size*, containing the maximum number of customers that can connect to a candidate store has size j. After generating these arrays, we use Python and IBM's CPLEX to create the linear program (Equation 8.1 to Equation 8.5) and add the variables and constraints with a 1:1 mapping between the generated arrays and the linear program. Therefore, we can store each variable regardless of the redundancy. Then, we run the linear program, which outputs which facilities are open and which customers connect to them. Using this and the cost matrices, we can determine the total system cost.

Before considering the results, however, we shall now return to the question of selecting candidate store locations. Visualize the locations of the customers as points on an *x*-*y* plane. Then, we draw lines upon this plane and select the center of each grid as a candidate store location. If we draw these lines close together, say, every 50 meters, then we have a large number of candidate store locations, which dramatically increases the computational and space complexity. However, if we draw the lines very far apart, say, ever 5000 meters, then we have a small number of candidate

store locations, and potentially limit the solutions by excluding the possibility of a large number of small facilities. We choose to draw lines every 500 meters. In addition, we aggregate our population data into 50-meter by 50-meter squares, again selecting the center of each square as the location and assigning each location to be the number of customers in the area.

After generating this aggregated data, we compute a matrix containing the distance between each customer and each candidate store location. From this, we can compute the arrays, as described above, but with one permutation: we treat each aggregated customer location (the center of the 50-meter by 50-meter square) as a single customer, then, after the various costs have been computed, we multiply the connect cost by the number of customers in a region. This permutation also requires an additional array, *customers*, which has size i and represents the number of customers in a given aggregated area. This new array is required because we need to check that a store is not connected to too many customers (6). As such, we must modify the constraint to account for the fact that the variable c_{ij} may actually represent several customers:

$$\forall j : \sum_{i} x_{ij} * s_i < q_f \qquad \text{Equation 8.12}$$

Where s_i represents the number of customers in an aggregated area i. In summary, we take population data, aggregate it so that we have lower resolution to decrease the computational and space complexity, then compute the various costs for customers to drive to facilities, trucks to drive to facilities, and the facilities to open. Then, we run the following linear program:

Minimize:

$$\sum_{i,j} c_{ij} x_{ij} + \sum_{j} f_j y_j \qquad \text{Equation 8.13}$$

Subject to:

$$\forall i, \forall j : y_j - x_{ij} \geq 0 \qquad \text{Equation 8.14}$$

$$\forall i, \forall j : x_{ij} \in \{0,1\} \qquad \text{Equation 8.15}$$

$$\forall j : y_j \in \{0,1\} \qquad \text{Equation 8.16}$$

$$\forall j : \sum_i x_{ij} < q_f \qquad \text{Equation 8.17}$$

$$\forall f : \sum_f y_{\{(j \,\%\, length(f))*f\}} \leq 1 \qquad \text{Equation 8.18}$$

$$\forall j : \sum_i x_{ij} * s_i < q_f \qquad \text{Equation 8.19}$$

After running this program, we analyze the resulting partitioned network.

8.2.4. Results of solving the designed green supply chain and the retail store density problem

We used the same parameter estimations as Cachon (Cachon, 2014)uses, in order to compare our results. These parameters are summarized in Table 8.2 and Table 8.3.

Table 8.2. Parameter estimations

	Cars	Trucks	Stores
v	0.0804	0.4840	4.09
f	0.1110	0.3920	-
p_f	0.980	1.050	0.44
e	2.325	2.669	2.43
q	18	20000	141

The car's q value is the demand for the system is the amount that a customer purchases in a period. The store's q value is the 'packing efficiency', in units' kilograms of products per square meter. The period for this system is one week, that is, customers are assumed to shop at the store once a week, trucks resupply the store once a week, and store costs are the cost for one week of operation. These parameters are then applied to the cost functions defined above (8, 9, 11), with a distance of 1 km for the cars and trucks and a period of 1 week for the stores.

Table 8.3. Cost functions applied to parameter estimations

	Car ($/ km)	Truck ($/km - customer)	Small Store ($/ customer-period)	Medium Store ($ / customer-period)	Large Store ($/ customer-period)
Operating	2.1e-02	1.6e-03	1.77	8.89e-1	7.48e-1
Emissions	2.87e-05	1.89e-06	9.46e-4	4.75e-4	4.00e-4

To calculate the parameters in Table 8.3, all stores are assumed to be at max capacity, and prices emissions at $1 / ton CO2. For input, we used a high-precision population density map of Puerto Rico, which provides population density at a resolution of 30 meters by 30 meters. Data is stored as a 3-tuple of the x-y coordinates of each location and the number of people living at that location [116]. We analyzed two subsets of this data. The first contains the entire island, which has an area of 9,104 km^2 (170 km by 60 km), and a population of 3,994,259 people. The second contains the city of San Juan, Puerto Rico's largest city and capital. We considered both the city and the surrounding area, sub-setting a region with an area of roughly 2,500 km and a population of 2,199,923. These two regions are seen in Figure 8.5.

We subset this data into roughly 1,000 customer nodes and 300 store nodes, in order to keep the complexity low enough for the problem to be solvable in a reasonable period of time. For the whole island scenario, there are 1155 customer nodes with area 1155 km^2 and 554 store nodes with area 20.36 km^2. For the San Juan simulation, there are 1017 customer nodes with area 2.26 km^2 and 279 store nodes with area 9.0552 km^2. These nodes are then used to calculate cost matrices, which form the input for the integer program as described in (13-20). This was solved using IBM's CPLEX. Simulations were run on a high-end desktop computer, with up to 5 hours allotted for each run. Each simulation ended after either running for 5 hours or achieving 1% optimality. All simulations reached 3% optimality, so values can generally be considered accurate to $\pm$3%.

We consider two baseline scenarios: operating cost minimization and emissions minimization. We ran the above linear program over the two data sets, and found that, in the whole-island scenario, $n_o = 134$ and $n_e = 260$, indicating that the emission-minimizing system has many more, smaller stores that are closer to customers (n_o represents the number of stores in operating cost minimizing system and n_e represents the number of emission minimizing system). This result is in line with the result of earlier research, notably Cachon (Cachon, 2014), in that the emission minimizing system has more stores than the operating cost minimizing system. Plotting the x-y locations of customers as ***circles*** with the color denoting the store they connect to, and stores as ***squares*** with sizes according to their capacity yields

. Distances are in kilometers, and customers are scaled according to the number of people at the node. The central warehouse is not shown. The size of the stores is shown in three categories of small, medium and large.



Figure 8.5. Municipalities of Puerto Rico (Wikiland, 2020) .

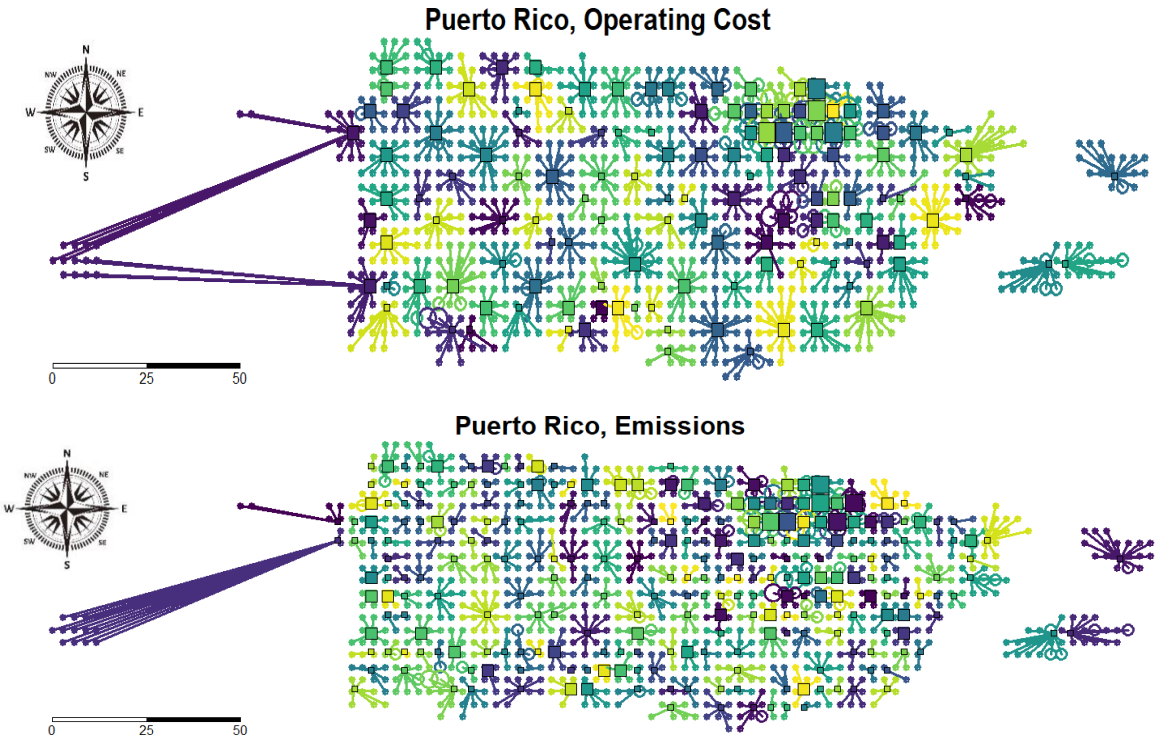


Figure 8.6. Locations of customers (circles) and the stores (squares) that they connect to, in the emission minimizing system and operating cost minimizing systems. Distances are in kilometers (Williams et al., 2020).

The fact that, in our model, $n_o < n_e$ (the emissions minimizing system has more stores than the operating-cost minimizing system), largely confirms the results of Cachon (Cachon, 2014) and others. The emissions-minimizing system likely has more stores because the cost of a customer driving to a store is so high. It far offsets any emissions or cost increases from additional stores

opening and trucks driving to them. This result is likely due to three factors: (1) trucks can carry goods for many customers, but their emissions and cost are much less than that of the customers, per kilometer (by a factor of 2). (2) No matter which store the customer connects to, trucks are required to supply the store, so that cost is somewhat fixed, making the cost that the customer incurs more variable. (3) Larger stores are more space-efficient than smaller stores and require less heating and cooling per unit of capacity. However, the difference between a small store and a large store is small compared to the cost of the distance that a customer drives. A customer driving an extra forty kilometers offsets the difference between a small and medium store in the operating model, and in the emissions model, just sixteen extra kilometers offsets the difference. This difference likely accounts for the different store layouts (the different number and arrangement of stores) in the operating-cost minimizing and emissions-minimizing models. We consider two forms of analysis: *penalty bound* and *gap reduction*. The emissions or operating penalty bound is the amount that emissions (or operating costs) increases when it is not the cost being minimized. In other words, the penalty bound is the amount of possible reduction, expressed as a percentage of the minimum cost. Then, the emission penalty:

$$P^e = \frac{C^e(o) - C^e(e)}{C^e(e)} \qquad \text{Equation 8.20}$$

Where P^e stands for the emission penalty, $C^e(o)$ and $C^e(e)$ represent the emissions cost C^e of an operating-cost minimizing system o and the emissions cost of an emission minimizing system e, respectively. Similarly, the operating penalty is:

$$P^o = \frac{C^o(e) - C^o(\mathrm{o})}{C^o(o)} \qquad \text{Equation 8.21}$$

For the baseline, whole island scenario, $P^e = 7.35\%$, and $P^o = 7.29\%$. In the San Juan scenario, $P^e = 5.55\%$ and $P^o = 7.84\%$ These results are striking and differ significantly from Cachon's (Cachon, 2014) estimated 0.3% for both values. This result is because of the change in the network topology – Cachon (Cachon, 2014) suggests that his model has a similar ratio of the total cost of cars and trucks, so the network is somewhat balanced between emissions and operating costs. That is to say, because a retailer in Cachon's model (Cachon, 2014) wishes to minimize the distance cars have to travel and trucks have to travel, and they want to minimize these distances roughly equally (a kilometer in one is about equal to a kilometer the other), the system that minimizes emissions is similar to the system that minimizes operating costs. Another reason for this difference comes from the motivation for using the k-median algorithm – under the TSP, a truck may deposit all of its goods at the first store, then travel to all the other stores while empty before returning to the warehouse, wasting resources and driving up the truck cost. Therefore, if a retailer is of a sufficient size such that it requires more than one truck every period, then trucks travel less distance if they go directly to the store and back. Since $q_t = 20000, q_c = 18$, 1111 customers need to connect to a store in order to require an entire truck. For the whole-island and San Juan scenarios every store meets this criterion, in both the emission and operating cost models.

As an aside, this result (that there are more stores in the emission minimizing scenario) is due to the higher cost of car transportation than truck (Table 8.3). However, in both scenarios, the cost of cars dominates, and so the penalty is relatively small, driven by the relative cost of cars and trucks. The gap reduction is considered the amount that any "hybrid" system reduces the emissions or operating cost relative to the total amount of reduction that can occur. Consider the emissions gap reduction, under three scenarios: operating cost minimization, emission minimization, and a hybrid of operating costs and some price of carbon. Then, the emission minimizing system should

have the least emissions, and the hybrid model should have the second least. We hypothesize that the operating cost minimizing system has the largest emissions, because the cost of those emissions is not built into the objective function, and so they cannot be minimized. The emissions gap reduction G^e of a hybrid system h is, therefore:

$$G^e(h) = \frac{C^e(o) - C^e(h)}{C^e(o) - C^e(e)} \qquad \text{Equation 8.22}$$

Similarly, the operating cost gap reduction $G^o$is:

$$G^o(h) = \frac{C^o(e) - C^o(h)}{C^o(e) - C^o(o)} \qquad \text{Equation 8.23}$$

Considering a carbon tax of $100 / metric ton of CO2 (Eisaman et al., 2018), in the whole-island baseline scenario of our model, the maximum possible reduction in emissions is 7.35%. $G^e(h_{100}) = 21.08\%$, indicating that a carbon tax of $100 / metric ton of CO2 achieves 21% of the total emissions reduction possible, or a 1.54% reduction. Given that $100 / metric ton a rather high price, this indicates that a carbon tax is relatively ineffective in encouraging a greener supply chain. In the San Juan scenario, this gets even worse - $G^e(h_{100}) = -6.18\%$, indicating that the added carbon tax actually resulted in increased emissions. Table 8.4 summarizes the values of G^e and G^o for the San Juan and whole island scenarios.

All values are percentages, so $G^o > 100$ means that the operating cost was lower in the carbon tax scenario than in the operating-cost minimizing scenario. In the San Juan scenario, this result is interesting, and counter intuitive. These results are due to the limits of the region—in the whole island scenario, the effects of each change to the scenario are easier to see, because it's a larger area.

Table 8.4. Carbon tax gap reduction.

	Whole Island		San Juan	
Carbon Tax	*G^o*	*G^e*	*G^o*	*G^e*
50	95.49	5.73	106.57	-6.18
100	96.40	21.08	104.15	-10.13
500	91.39	53.32	107.09	41.80

In the San Juan scenario, the relatively small region, combined with the fact that CPLEX may not find the optimal solutions (due to the time and complexity limitations), results in a decrease in precision. However, the San Juan scenario is useful, because it indicates what may happen in smaller regions, like cities—that is, even if a carbon tax were effective on large supply chains, if it does not work in smaller regions, then it is not effective at reducing emissions. The San Juan scenario shows that our model indicates it is even harder to drive down emissions in smaller regions, like cities. This is because retailers may only consider city-level changes to their supply chain, and likely do not have a single warehouse for a very large region. Considering the San Juan scenario addresses this concern, because the limits of different scenarios is shown in a smaller region. In the gap reduction measurement, small changes in values are over-emphasized, and for this reason, the operating cost and emission values for San Juan example are shown in Table 8.5.

The difference between the base operating cost scenario and the hybrid scenarios is very small (<3%) and attributed to not reaching the optimal solution and the limited solution space within the confined region. To further demonstrate this, consider in Figure 8.7, which graphs of San Juan and minimizing emissions, operating cost, and the hybrid (carbon tax) scenarios are shown. Distances are in kilometers, and circles (customers) are scaled according to the number of people at the node. The central warehouse is not shown.

Table 8.5. Operating cost and emission values.

	Whole Island		San Juan	
Scenario	C^o	C^e	C^o	C^e
Operating Cost ($)	9973481	9612	4025020	3563
Carbon Tax $50/ton CO_2	10006236	9575	4004306	3575
Carbon Tax $100/ton CO_2	9999607	9473	4011917	3582
Carbon Tax $500/ton CO_2	10036084	9261	4002663	3485
Monetized Emission ($)	10700667	8954	4340453	3376

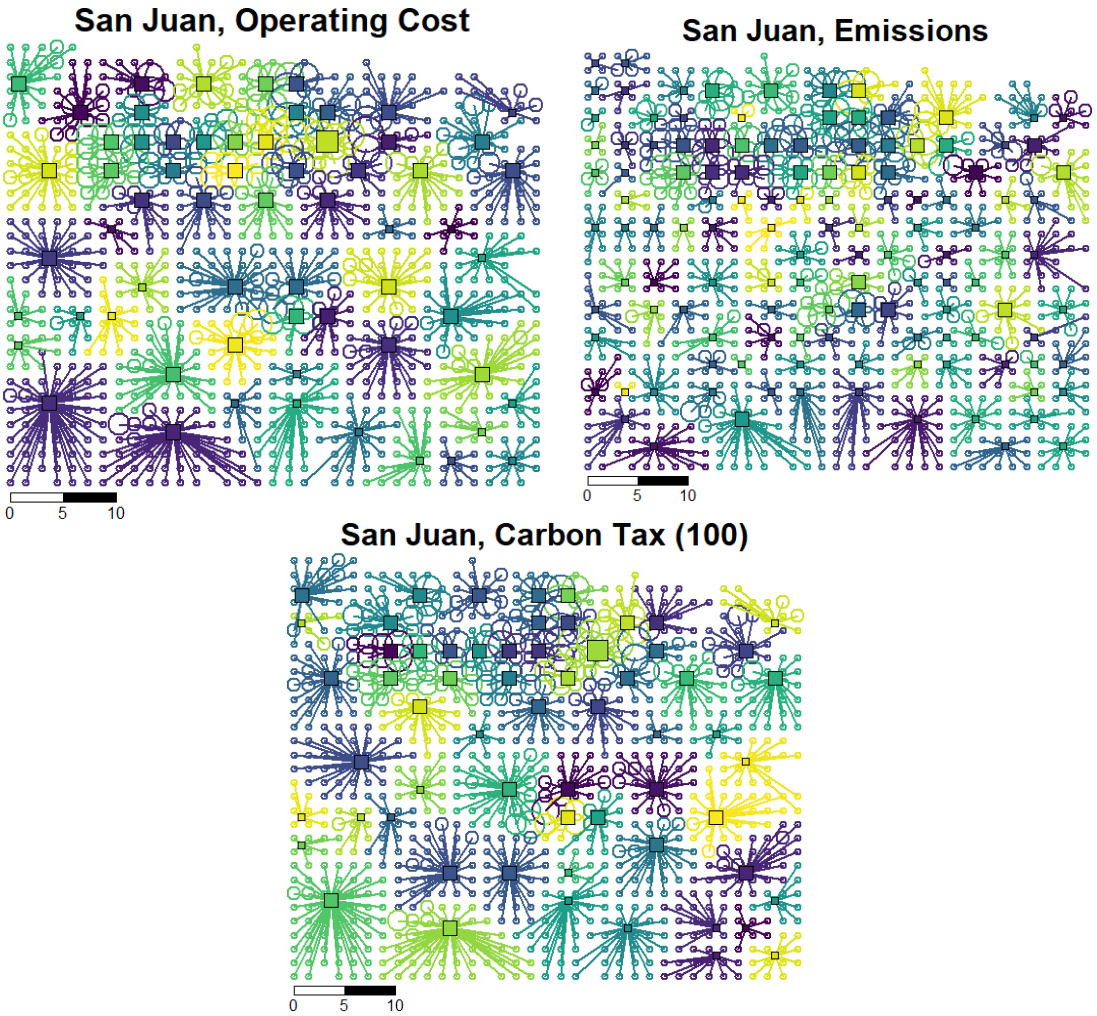


Figure 8.7. San Juan operating cost, carbon tax, and emission minimizing scenarios.

The size of the stores is shown by the size of the squares in three categories of small, medium and large. Finally, we consider eleven possible scenarios and compared the effect they have on the baseline, as shown in Table 8.6. To facilitate the comparison with previous studies, the same parameter estimates as Cachon (Cachon, 2014) are used.

In Table 8.6, values are averaged from the two scenarios. Costs are measured relative to the baseline. Negative values indicate increases and are underlined for emphasis. Doubled Gas Prices doubles the cost of fuel for cars and trucks.

Table 8.6. Different scenarios

	Operating Cost		Emission	
Name	Reduction	P^e	Reduction	P^o
Baseline	-	6.45	-	7.57
Doubled Car fuel efficiency	14.02	2.03	29.05	2.29
Doubled Truck fuel efficiency	0.32	7.20	0.34	7.51
Doubled Car, truck efficiency	14.50	2.40	29.31	3.23
Doubled Fuel Prices	-22.60	2.19	**0.00**	2.77
Low electricity emissions	**0.00**	19.80	27.99	20.22
High electricity emissions	**0.00**	1.59	-20.48	2.43
High rent cost	-44.12	16.27	**0.00**	22.55
Low electricity emissions, high rent	-44.12	39.16	27.99	41.89
High store fuel usage	-7.51	0.69	-56.59	-0.36
High store fuel usage, high rent	-5.17	2.38	-56.59	3.97

The sensitivity analysis conducted using these scenarios, demonstrates that our model is resilient to even large changes in the input parameters. This implies that a supply chain is built

under either an operating-cost minimizing, or emission-minimizing strategy is similar to a supply chain under a different set of input parameters. This resilience is expected, both from previous research and from the opposing costs of customer's cars and retailer's trucks and stores. Cachon (Cachon, 2014) found a similar gap reduction for many of these scenarios. In the baseline, scenario, the penalty for choosing one scenario over another is somewhat small (~7%), and so a system that minimizes the operating cost ends up looking similar to a system that minimizes emissions. This is because the largest cost, under both systems, is the cars (Table 8.2). The difference between the two models is then due to the exact balance point between the customer's and the retailer's costs.

Four scenarios in Table 8.6 have identical performance to the baseline. For instance, the high store fuel cost scenario increases emissions, while the high rent scenario cost does not. This is due to the fact that the operating cost / emissions reduction is measured relative to the respective minimizing scenario – in the high rent, operating-cost minimizing scenario, the emissions are increased over the emissions in the baseline, emission minimizing scenario (that is, $P^e_{high\ rent} > P^e_{baseline}$). However, because the rent does not contribute to GHG emissions, it is not considered by the emission minimizing scenario, resulting in the high rent, emission minimizing scenario being the same as the baseline emission minimizing scenario. This is the same for the doubled gas price scenario. The converse holds for low and high store electricity emissions–the operating cost does not change, although the emissions do.

Note in the doubled gas prices scenario, the penalty (P^e, P^o) is more than halved, indicating that the operating-cost minimizing system is approaching the emission minimizing system, so while the operating cost has increased, the emissions of the system have decreased. In this case, doubling the gas prices is actually more effective than a carbon tax of $100 / metric ton CO2, which has $P^e = 5.80$ (note that these are directly compared because they are relative to the same

baseline). This is because doubling the gas price increases the cost of both trucks and cars (but, importantly, not the cost of opening more stores), and increases the cost of cars more than the cost of trucks (it increases the car operating cost by 57% and the truck operating cost by 46%). Therefore, the operating-cost minimizing system have more overall stores, which in turn means fewer overall emissions.

In six scenarios in Table 8.6, either operating cost or the emissions of the system (underlined) are increased. In the case of high store electricity emissions, high rent cost, and high store fuel cost (operating cost), this is expected. The high store fuel cost, emission minimizing scenario, however, is interesting. In this case, the operating cost rose only mildly (7.51%), but the emissions increased by 57%. This, again, shows the impact of non-fuel costs—only a small portion of the store's operating cost controls the emissions of the store. Increasing the store's fuel consumption (which models a store with inefficient heating and cooling system) barely raises the operating cost, but drastically increases the emissions.

Based on the results, the best way to decrease the system's emissions is to double the fuel efficiency of cars. Cachon (Cachon, 2014) concluded that doubling the fuel efficiency of trucks would have the most significant effect, however, this is likely due to a different topology. In his model, a change to the truck's fuel efficiency could propagate down and make a simulation with a higher number of stores more feasible, but in our model, this effect is somewhat limited by the already low cost of trucks. Moreover, Cachon's (Cachon, 2014) recommendation are based on the emissions gap reduction per system – that is, in a system with doubled truck fuel efficiency, a carbon tax is more effective in reaching the minimum emissions for that system. However, according to our results, even the best-case scenario for doubled truck fuel efficiency is much worse than that of doubled car fuel efficiency, making it a less viable option. Another viable

scenario is doubling gas prices, which would have a similar effect to doubling car fuel efficiency in the operating cost minimizing scenario, however, it has no effect on the emission minimizing scenario. Therefore, while it is effective in creating a greener system, other scenarios (like doubling the car or truck fuel efficiency or low retailer electricity emissions) lead to overall less emissions even though the overall emissions penalty may be higher. In other words, doubling the price of fuel encourages a supply chain design that is closer to a greener supply chain (that is, it has more stores), while other scenarios change the amount of emissions of the system while not necessarily changing the supply change design – in the operating cost minimizing system.

8.2.5. Take aways of solving the designed green supply chain and the retail store density problem

At the beginning of this section, the importance of considering emission reduction goals in design to cope with global warming is reinforced. Also, it is argued that to design a GSC that emits much less GHG emissions than classic SCs, policies should incentivize increasing energy efficiency. Based on a critical evaluation of the literature, using TSP for vehicle routing and store location, single transportation mode, hierarchical SC design, fixed-location store location, classic vehicle routing using pre-defined demand distributions are found as the main gaps in designing a GSC. Accordingly, removing the unreal assumption of the Traveling Salesman Problem (TSP) – starting and ending at the same location – we use an alternative k-median based SC network partitioning, considering various candidates for the locations of the facilities and using actual demand-based vehicle routing instead of predefined distribution-based models. These are the main contributions that we make by our proposed design approach.

Our findings confirm the results of previous literature including Cachon (Cachon, 2014) that the emission minimizing supply chain has more stores than the operating cost minimizing supply

chain, and that the penalty bound between the two is roughly equal. However, it is found that the penalty bound is much larger than estimates in the previous studies. Additionally, while others conclude that doubling the fuel efficiency of trucks is the most effective in creating a greener system, we find that doubling car fuel efficiency lowers emissions far more than doubling truck fuel efficiency, and that doubling the cost of fuel decreases the penalty bound more than doubling truck fuel efficiency. Based on this result, it is demonstrated that doubling the cost of fuel creates a more robust system that is closer in design to the green supply chain, while doubling car fuel efficiency creates a greener system overall.

We also confirm the results of Cachon that a carbon tax is not effective in creating a greener system. We find that a carbon tax is much less effective than Cachon, creating minimal differences at reasonable tax amounts ($100 / metric ton). While in the abstract, an increase in the cost of emissions should drive emissions down, in cities with a limited set of options for where and how to place stores, this does not always happen. This is due to the fact that a carbon tax applies to all parts of the system – cars, trucks, and stores – when the cars are the biggest polluter. The carbon tax's increase to the cost of stores and trucks is effectively countering most of the increase to the cars, resulting in only a small shift in the overall balance between the three costs, which results in only marginally changing the balance between more or less stores. Therefore, a carbon tax is less effective than targeted solutions, like doubling the cost of fuel, which leaves stores out, or doubling car fuel efficiency, which addresses the root of the problem.

The proposed design approach is not limited to GSC design. It is easily extended to many design problems, including manufacturing, material design, health care, and energy transmission and distribution. The same principles in our proposed GSC design approach can be applied in selecting the cost-effective material supplier with less GHG emission material supplier,

manufacturing operation facility, heath care provider, and energy distributer in these networks. For example, in manufacturing, a k-median network partitioning model can be used to assign the facility to a group of operations, services and products considering operating costs and GHG emission reduction goals. Also, in material design process, the flow of the material can be modeled as green material selection problem, to minimize cost and guarantee product performance as well as reduce the entire life-cycle impact to the environment via GHG emission reduction.

8.3 Impact of Asset Management in a Green Supply Chain

With increasing concerns about global warming caused by greenhouse gasses (GHGs), organizations have become more responsible for their operations. According to the U.S. Environmental Protection Agency (EPA), companies with a supply chain (SC) generate about 42% of GHGs in their transportation (30%) and inventory systems (12%), which makes mitigating climate change through a green supply chain (GSC) management a reasonable solution.

To design a GSC, we model the SC as a customer and store network, with customers driving in cars to and from stores and the retailer resupplying the stores from a central warehouse. The number and location of stores are determined to find a low-cost and low emission configuration for the SC. The key findings are (1) SCs with more small stores generate less emission than ones with fewer large stores; (2) when minimizing the operating cost is more important than mitigating GHG emissions, fewer large stores are preferred than having more small stores; (3) a SC with two warehouses reduces the number of open stores in a large area such as Puerto Rico.

Our contributions are (1) building a model of a GSC based on population data;(2) modeling a GSC in a two-echelon network which can be solved simultaneously using the k-median approach;(3) evaluating the effect of multiple warehouses on the overall GHGs emissions; (4)

managing the incompleteness and inaccuracy of the data through implementing the compromise Decision Support Problem construct to identify satisficing solutions.

The model mentioned earlier highlights the important parameters that impact the green GHG emissions reduction from a SC that describe in this section. We also discuss how this approach can be employed for other design problems, including manufacturing and healthcare.

8.3.1. Frame of reference on asset management in a green supply chain

The gradual increase in the overall temperature of the Earth's surface, oceans, and atmosphere is called global warming. This is mainly due to human activities such as deforestation, farming, and fossil fuel consumption that release carbon dioxide (CO_2), methane, and other greenhouse gases into the atmosphere. Greenhouse gases have far-ranging environmental and health effects besides causing climate change. They result in respiratory illnesses and diseases through air pollution.

Global warming is a huge threat to the environment. Extreme climate, food supply disruptions, and increased wildfires are other effects of climate change caused by greenhouse gases. One way to mitigate global warming is to use our resources and conduct our activities in different sectors in a sustainable fashion. For example, the green supply chain is one way that can benefit our environment while helping companies and consumers save money. The supply chain has been traditionally labeled as a one-way, integrated manufacturing process where raw materials are transformed into final products, then delivered to customers (Mentzer et al., 2001).

The largest share of greenhouse gas emissions generate by the transportation sector in the supply chain since transportation and electricity use of a retailer's supply chain network are the main factors in determining its performance (Afshari et al., 2014; Elhedhli and Merrick, 2012;

Fahimnia et al., 2015; Harris et al., 2014; Mohebalizadehgashti et al., 2020; Waltho et al., 2019). Thus, the supply chain network is an interesting case in terms of GHG emission reduction. We are able to reduce environmental pollution and production cost through a green supply chain (Elhedhli and Merrick, 2012; Waltho et al., 2019) , and promotes economic growth, generates a competitive advantage in terms of greater customer satisfaction, positive image, reputation, and better export of their products in pro-environmental countries (Rehman Khan, 2018).

Companies of all sizes work in global industrial networks and global supply chains. They influence the environment due to their large supply chain and supplementary processes like transportation and packaging. Sustainability is vital for companies and enterprises to investigate their global supply chains through three spheres of sustainability (Environmental, Social, Economic). Impact on the environment deals with Environmental sustainability, while economic sustainability implies financial stability. Social sustainability deals with health and safety for people (Bhingea et al., 2015).

A range of topics has been discussed in the literature regarding green supply chain network design. A comprehensive review (105 articles) on green supply chain network design between 2010 and mid-2017 is presented in (Waltho et al., 2019). Models and methods that explicitly include carbon emissions and environmental policies are described in this reference. The adoption and impact of carbon policies on supply chain network design are also discussed there. Regarding suppliers with emission reduction goals, the problem of capacity is also vital. For instance, Walmart has promised to mitigate one billion metric tons of GHG emissions from its SC network by 2030 (Walmart, 2018a). These drops can happen from the retailer network and are often called Scope 1 GHG emissions.

Another example of the topic discussed in the literature is the capacity of stores. As discussed in (Williams et al., 2020), the capacity of retail facilities is of interest to city planners, and it is associated with several challenges such as car culture. Car culture is a society or way of life characterized by excessive use of motor vehicles. In the case of retail stores, developing larger stores located far from each other forces the customers to travel longer distances, resulting in more emissions. However, there are many other factors that retailers may consider locating their stores correctly. Such factors, for example, include their long-term plan or the sizes of competing stores in the same area. If all the retailer's stores' features are the same, except the size, a customer prefers the accessibility of a close store rather than a far one (Williams et al., 2020).

Our main goal in this section is designing GSC to identify the store locations and distribute resources to achieve less total cost of operation and GHG emissions (Williams et al., 2020). As mentioned earlier, transportation is playing one of the leading roles in the green supply chain. Hence, the location and number of the stores are essential keys in a green supply chain. Likewise, we should think about the replenishment truck as well. One class of vehicle routing problems studied extensively is the Traveling Salesman Problem (TSP). In TSP, it is assumed to route the trucks, start at a warehouse, visit each store, and return to the warehouse.

We consider k-median instead of TSP. In general, the goal is to partition a network (typically a complete graph of all points in the network) into k partitions. The distance between each point and its partition center is minimized. The main distinctions between TSP and k-median design are (i) the retailer can select the stores' locations and (ii) the retailer is responsible for the customers' travel expenses (for example, the "last mile" of the SC is not neglected) (Williams et al., 2020).

In (Williams et al., 2020) and the present work, the store locations are not pre-determined and are located as part of the solution. The retailer in our model has a single mode of transportation,

and the main focus is on distances traveled rather than lead time. Our main focus is on the downstream supply chain consisting of inbound replenishments to facilities, the facilities themselves, and the portion between the retail facilities and customers' homes. As the number of facilities grows, a dense network is created, facilities become smaller, and customers find themselves closer to a store; thus, they do not need to travel long distances for shopping (Williams et al., 2020).

Regarding the literature review, we noticed it is important for companies to investigate sustainability in their supply chain through three spheres of sustainability (Environmental, Social, Economic) to compete in the market that is possible through the green supply chain. There are various factors and methods to design the GSC, that we review some of them. In this section, we design a GSC to identify the store locations and distribute resources to achieve less GHG emissions and operational cost. This helps companies to manage their resources, reduce their environmental footprint and understand influential decision factors and asset management's impact on GHG emissions reduction.

In our model, the supply chain is modeled as a two-echelon network, in which goods flow from a single, central warehouse to stores, then from the stores to customers. Trucks transport the goods from the warehouse to the stores and cars from the stores to the customers. Neither locations and nor the number of facilities is pre-determined. Both vehicle routing problems (cars and trucks) are modeled as k-median problems. In the model, a retailer's operating cost consists of three components: cost of stores, trucks, and cars. The truck cost is the cost of a truck driving from a warehouse to the store. The car cost is the cost of a car driving from the customer's home to the store and returning home. The inventory cost is the cost of having a store large enough to fulfil all

customer's needs, the building's cost, and the heating and cooling cost of the building (in order to consider buildings GHG emission from the heating and cooling)

As mentioned earlier, the warehouse is centrally located. Therefore, for a large region such as Puerto Rico, the area is divided into two parts, and a central warehouse is considered in each part. Hence, each store is still connected to one warehouse and has a truck connecting directly to it from the warehouse. However, since distances between some stores and the warehouses are reduced, those may affect the cost of replenishment trucks and the cost of connecting customers to stores.

All customers are assumed to drive cars, and multi-modal travel (such as public transportation or walking) is not considered. The approach considered in this section is the same as the one presented in (Williams and Cremaschi, 2019). However, there are two main differences: (1) We formulate the problem as a compromise Decision Support Problem (cDSP) instead of using optimization and solve it with MATLAB; (2) We also consider the study region as two parts and assign two warehouses in the centre of each part instead of assuming a single warehouse in the middle of the whole region.

The remainder of the section is as follows: In Section 8.9 and 8.10, we clarify how and why we initiate our GSC design in cDSP. In Section 8.11, we discuss the results of a sensitivity analysis. In Section 8.12, we show the verification and validation of our work, and finally, we end with the closing remarks in Section 8.13.

8.3.2. Method used for asset management in a green supply chain

The decision support problem

In system theory, the components of a system may result in organized complexity. However, knowledge of the components alone may not be sufficient to predict the system's behavior. On the

other hand, computational complexities can arise from random and statistical variability of the components and the subsystems and systems they form. Thus, managing complexity and uncertainty are both essential to gain enough system knowledge (Smith et al., 2015). To model our GSC, we are using the compromise decision support problem(cDSP) construct. The requirements for the cDSP are identifying a set of solutions that are relatively insensitive to uncertainties. In cDSP, we are not looking for peak, optimum, or global solutions. We are looking for flat and relatively insensitive solutions to perturbations and to uncertain keys to things that we cannot control. Decision Support Problem (DSP) construct is based on the philosophy that design is fundamentally decision-making and model-based process (Smith et al., 2015).

In reflecting on the compromise DSP, parallels with the "demands" and "wishes" of Pahl and Beitz (Pahl, 2007) can be drawn. The demands are met by the satisfaction of the DSP constraints and bounds, and the goals represent the wishes. Thus, the feasible design space can be defined by constraints and bounds. The feasible and aspiration spaces together form the solution space (Smith et al., 2015).

Proposed word problem for asset management in a green supply chain

In the cDSP, the keywords "Given," "Find," "Satisfy," and "Minimize" are used in the form of the word formulation (Smith et al., 2015). The word formulations of our cDSP are as follows:

Given

- The number of customers. The number of store locations. The number of the store sizes.
- Cost of each customer to connect to a store for each pair of customer and candidate store location (Cij). (Is the cost of a customer driving to a store plus the truck's partial cost-driving from the warehouse to the store. It has considered the distance from the customer to the store

also store to the warehouse, the non-fuel variable cost for the car and truck to travel per unit of distance, the amount of fuel required per unit of distance, price of fuel per unit of fuel, amount of emissions due to the car and truck per unit of fuel (kg-CO_2 / L), price of emissions ($/kg-$CO_2$).

- The cost to open each store regarding the size (fj). (Is the cost, of a store *j* opening, including the building's cost plus the cost to heat and cool the building.)
- The maximum number of customers that can connect to the candidate store with size j (s_j).

Assumptions for the decision support problem for asset management in a GSC

- We assume that customer shopping, truck replenishment, and store cost operations happen once a week.
- Prices emission at \$1/ton CO_2. All customers are assumed to drive cars directly from their house to the stores and back. Note that we do not model additional complexity due to road networks. All stores are at max capacity.
- Each store must connect to a single warehouse. The warehouse is centrally located in the region.
- For comparing our results of one warehouse versus two warehouses, we divide the region into two different parts., east and west, and execute our program two times with our data and consider one central warehouse in each region. We then combine our results for both parts (number of customers, number of candidate stores, number and size of open stores). Note that we do not consider two different warehouses at the same time in the region.

Find

Independent System Variables: (they describe the attributes of an artifact).

- x_{ij} is a binary tracking variable if costumer i connecting to store j.
- y_j is a binary tracking variable if store j is open.

The Deviation Variables*:* (they indicate the extent to which the goals are achieved).

- d_1^- is underachievement of the system goal with minimized GHG emissions.
- d_1^+ is an overachievement of the system goal with minimized GHG emissions.

  Where $d_1^+ \cdot d_1^- = 0$ and d_1^+ and $d_1^- >= 0$

- d_l^- is the lowest number of customers are connected to the store l-1 than the maximum capacity of that store.
- d_l^+ is the highest number of customers are connected to the store l-1 than the maximum capacity of that store.

  Where $d_l^- . d_l^+ = 0$ and $d_l^- and\ d_l^+ >= 0$

Parameters

K Set of indices used for store sizes.

c_{ij} Cost of customer i connecting to store j.

f_j^k Cost of opening store j of size k.

q_j^k The capacity of store j of size k.

x_{ij} Binary variable indicating if customer i is connected to store j.

y_j^k Binary variable indicating if store j of size k is open.

s_i Represents the number of customers in an aggregated area i.

LB Lower bound of our system.

UB Upper bound of our system

J Total numbers of candidate stores in the region under consideration.

l Refers to the number of the system goal. Its boundary is between 1 to J +1.

d_i^- and d_i^+ Deviation Variables indicate the extent to which the goals are achieved.

Satisfy

System Constraints: (Must be satisfied for the solution to be feasible.

- Every customer must be connected to exactly one store.
- No customer is connected to a store that is closed.
- Each candidate store location must be modeled as a collocated candidate location with the additional constraint.

System Goal: **(**Must achieve a specified target value as much as possible.

- It is desirable to achieve zero GHG emissions when we consider the green supply chain.
- The number of customers who could connect to a store is limited, so the store's size and customers' size are important.

Bounds on the System Variables

- LB → The system variables should not be less than a specified value that is, our lower bound. The "LB" is shown in equations (6) and (7)
- UB → The system variables should not exceed a specified upper limit that is, upper bound. The "UB" is shown in equations (6) and (7)

Minimize Deviation Function

Minimize the system goal's overachievement with minimized GHG emissions and minimizing the highest number of customers who exceed the maximum capacity of store j.

Proposed Design Approach

We formulate the Retail Store Density Problem as a cDSP method. Let k refer to the set of indices used for store sizes, f_j^k refers to the cost of opening store j of size k, c_{ij} refers to the cost of a customer i connecting to store j, x_{ij} is a binary variable tracking whether customer i connects to store j and y_j^k is a binary variable indicating whether store j of size k is open, s_i represents the number of customers in an aggregated area i, q_j^k refer to the capacity of store j of size k (In our problem, q^k, is the same for all j) and d_l^- and d_l^+ refer to deviation variables that indicate the extent to which the goals are achieved. The proposed approach employed here is shown in Figure 8.8. As we mentioned earlier, we need to satisfy the system constraints to make the solution feasible. The system goals must achieve a specified target value as much as possible and the upper and lower bounds on the system variables and deviation variables.

We formulate the Retail Store Density Problem as a cDSP method. Let k refer to the set of indices used for store sizes, f_j^k refers to the cost of opening store j of size k, c_{ij} refers to the cost of a customer i connecting to store j, x_{ij} is a binary variable tracking whether customer i connects to store j and y_j^k is a binary variable indicating whether store j of size k is open, s_i represents the number of customers in an aggregated area i, q_j^k refer to the capacity of store j of size k (In our problem, q^k, is the same for all j) and d_l^- and d_l^+ refer to deviation variables that indicate the extent to which the goals are achieved.

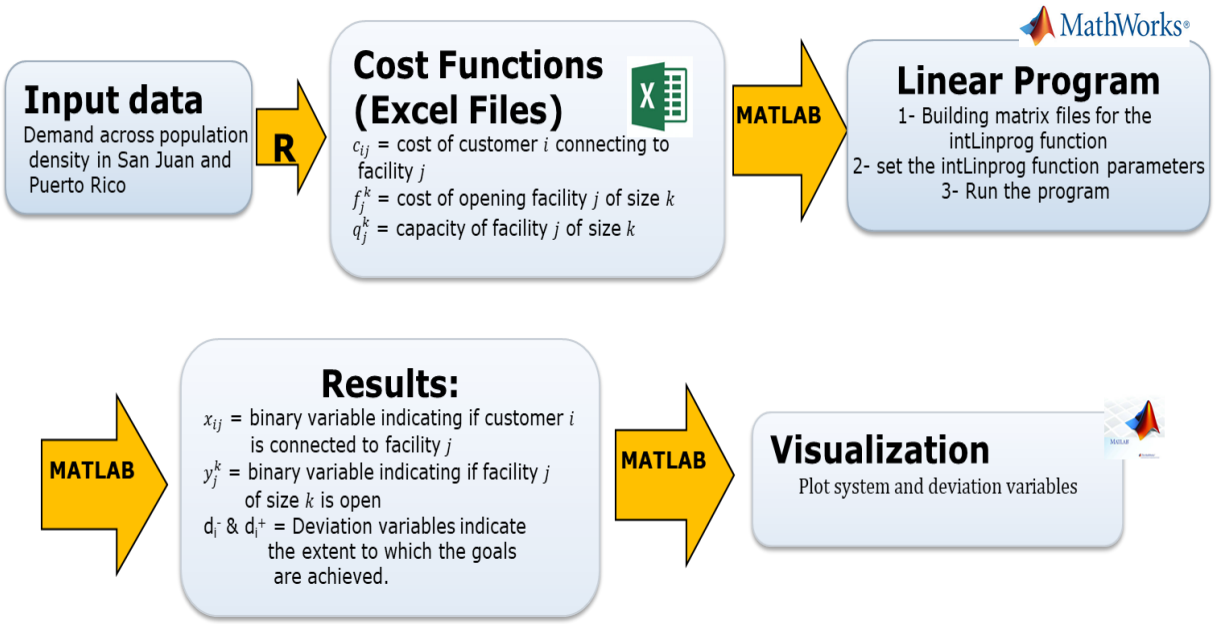


Figure 8.8. Solution method.

Equation 8.24 is considered to show that every customer must be connected to exactly one store. Equation 8.25 implies that no customer is connected to a store that is closed. Equation 8.26 implies that each candidate store location must be modeled as a collocated candidate location with the additional constraint (in each specific area, we should have just one size store determined by the number of customers aggregated in the area).

System Constraints

$$\forall i : \sum_j x_{ij} = 1 \qquad \text{Equation 8.24}$$

$$\forall i, \forall j : y_j - x_{ij} \geq 0 \qquad \text{Equation 8.25}$$

$$\sum_k \sum_j y_{k*j} \leq 1 \qquad \text{Equation 8.26}$$

System goals

$$\sum_{i,j} c_{ij} x_{ij} + \sum_j f_j y_j + d_1^- - d_1^+ = 0 \qquad \text{Equation 8.27}$$

where $d_1^- \cdot d_1^+ = 0$

$$\forall j : \sum_i x_{ij} * s_i + d_l^- - d_l^+ = q_f \qquad \text{Equation 8.28}$$

where $d_l^- \cdot d_l^+ = 0$

In Equation 8.27, d_1^- indicates underachievement of the system goal with minimized GHG emissions and d_1^+ indicates overachievement of the system goal with minimized GHG emissions. Equation 8.28 implies that each store must have fewer customers than some maximum capacity q_f where s_i represents the number of customers in an aggregated area i, d_l^- (l refers to the number of our system goal) represents the lower number of customers are connected to the store j (j = l-1) than the maximum capacity of the store, and d_l^+ represents the higher number of customers are connected to store j (j=l-1) than the maximum capacity of the store.

Bounds:

$$\forall i, \forall j : x_{ij} \in \{0,1\} \rightarrow \text{LB: } x_{ij} \geq 0 \quad \text{UB: } x_{ij} \leq 1 \qquad \text{Equation 8.29}$$

$$\forall j : y_j \in \{0,1\} \rightarrow \text{LB: } y_j \geq 0 \text{ UB: } y_j \leq 1 \qquad \text{Equation 8.30}$$

$$d1- \geq 0 \quad d1+ \geq 0 \quad d_l^- \geq 0 \quad d_l^+ \geq 0 \qquad \text{Equation 8.31}$$

$$\text{where } d1^- \cdot d1^+ = 0 \qquad d_l^- \cdot d_l^+ = 0 \qquad \text{Equation 8.32}$$

Equation 8.29 and Equation 8.30 constrain x and y to be binary and Equation 8.31 indicates that deviation variables should be positive and, at least one of the variables in each goal function is zero, which is ensured by Equation 8.32.

Minimize:

The compromise DSP's objective is to minimize a function that is expressed using only the deviation variables. This function is known as the deviation function. The deviation function is a representation of the deviation between the feasible solution space and the aspiration space. By minimizing this deviation function, the most robust regions of the solution space can be achieved. The range of the deviation variables depends on the goals themselves. However, the level of importance linked with achieving each goal varies for a designer. Hence, the goals are assigned weights, w_i, to affect a solution based on a designer's preference. These weights are usually normalized so that the sum is one. The general form of the deviation function, form system goals, in the Archimedean form, is as follows (Mistree et al., 1992):

$$Z(d^-, d^+) = \Sigma (w_i^- d_i^- + w_i^+ d_i^+) \qquad \text{Equation 8.33}$$

$$i=1,..., M \quad \text{and} \quad \sum_{i=1}^{m} w_i = 1$$

Here M is the number of the system goals. Regarding Equation 8.27 and Equation 8.28, we have J+1 number of system goals (J represents the total number of candidate stores in the region under consideration). l represents the number of our system goals. Hence d_l^- and d_1^+ are the deviation variables of system goal l. We define our deviation function as follows:

$$Z(d^-, d^+) = \Sigma (w_l^- d_l^- + w_l^+ d_l^+) \qquad \text{Equation 8.34}$$

$$l=1,..., J+1 \quad \text{and} \quad \sum_{i=1}^{l} w_l = 1$$

The weights, w_1, w_2, ..., w_l, reflect the level of desire to achieve each of the goals.

8.3.3. Proposed mathematical model to solve asset management problem in a GSC

The same GSC network design model proposed in (Williams et al., 2020) is employed in this section and shown in Figure 8.9. Orange line segments show connections between stores and a

single central warehouse. The blue circle represents the first echelon, and the red rectangle signifies the second echelon. Thus, three arrays can result from the GSC network, including *connect, open, size.* (1) *connect* array: this array contains each customer's cost to connect to a store.

For each pair of customer and candidate store locations, connect array has the size of $i * (j * k)$, where i is the number of customers, j is the number of store locations, and k is the number of store sizes. (2) *open* array: this array expresses the cost to open each store. In practice, this can differ for each candidate store because there may be different costs for each location. We do not model such scenarios and consider the price to open a store of a given size to be uniform.

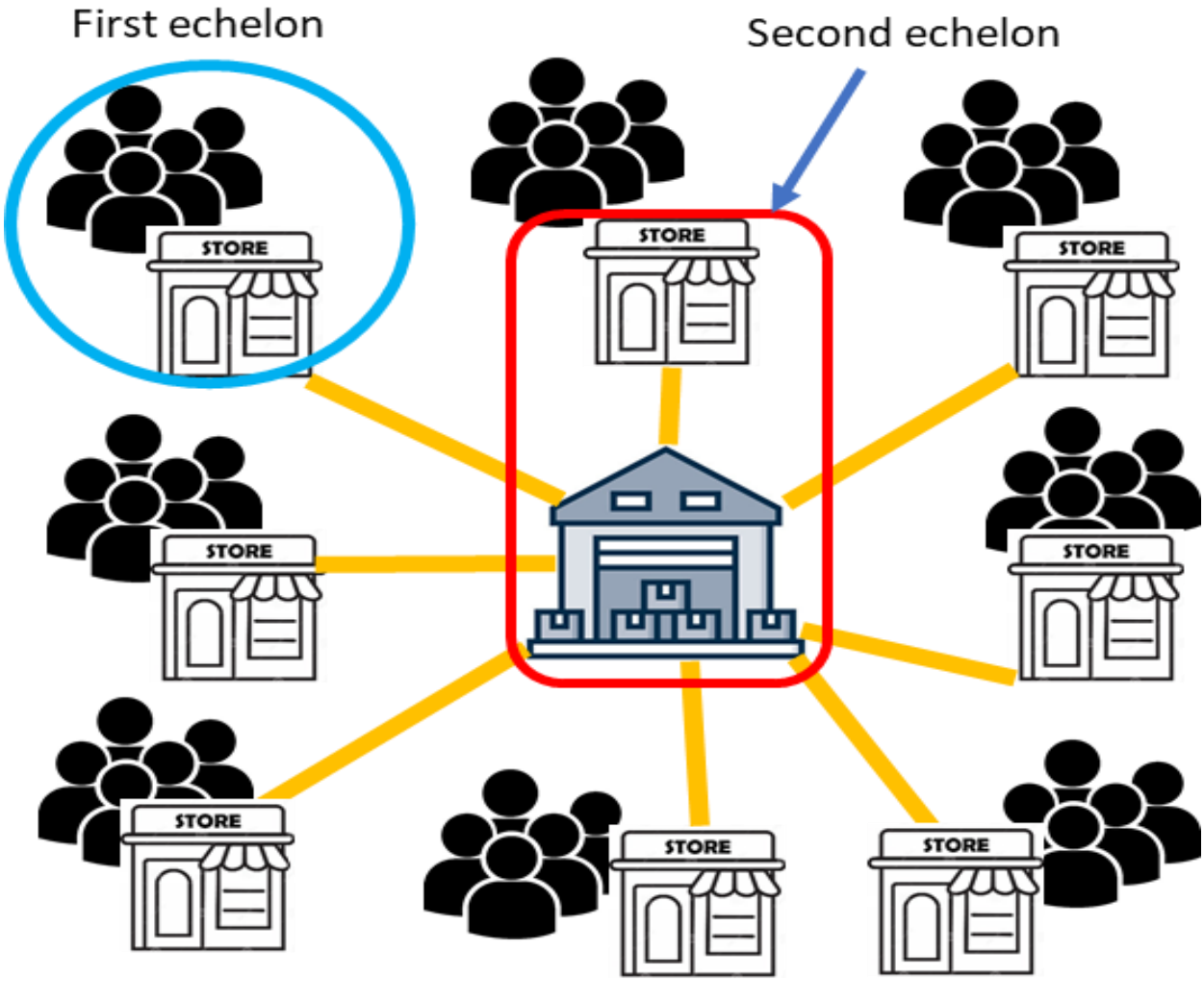


Figure 8.9. A two-echelon k-median network

Our input data is demand across population density in Puerto Rico. High-resolution (30 m x 30 m) population density data is available at (https://ciesin.columbia.edu/ data/hrsl/). Our data includes the cost of customer *i* connecting to store j (c_{ij}), the opening cost of each store (f_j), the number of customers in an aggregated area (s_i), and the capacity of the store regarding the size. After generating this aggregated data, we compute a requisite matrix to run the cDSP described earlier. Then we analyze the resulting partitioned network.

8.3.4. Results and discussion on the solution of asset management problem in a GSC

One week is considered as the system's operation cycle. This means that the period of calculation is for one week. In this period, customers are assumed to shop at the store once; trucks resupply the store once. As an input, we use a high-precision population density map of Puerto Rico, which provides population density at a resolution of 30 meters by 30 meters. Three different scenarios for the system are considered and analysed to evaluate the sensitivity of the results. The first case contains the entire island with one warehouse in the middle of the region. The second one is all Puerto Rico, with two warehouses on the east and west sides of the region. The third scenario is for the whole island by evaluating the different locations for the single warehouse. We considered each region's longitude and latitude and use the R program to determine the desired specific area's data sets. The following longitude and latitude are used for the analyses here:

a) Puerto Rico (whole island) longitude and latitude:

   from (-67.95867, -64.68755) to (17.67003,18.51725)

b) Westside of Puerto Rico longitude and latitude: from (-67.95867, -66.5) to (17.67003,18.51725)

c) East side of Puerto Rico longitude and latitude: from (-66.5, - 64.68755) to (17.67003,18.51725)

d) San Juan longitude and latitude: from (-66.11, - 66.20) to (18.305639,18.460065)

There are 1155 customer nodes and 990 candidate stores (330 candidate nodes) in three different sizes, small, medium, and large, for the whole island scenario with one warehouse. To calculate the whole island with two warehouses, the island is divided into two sections: west and east (with the latitude as mentioned earlier and longitude for each side). The program is run with data regarding each part and considering the single central warehouse in each region (

Figure 8.10) and then combined the results for both sections. Simulations for the whole island with one warehouse were run on a high-end desktop computer, with up to 7 hours allocated for each run. The location of the warehouse (the black triangle) considered for the whole of Puerto Rico is shown in Figure 8.11.

Figure 8.10. Candidate stores and warehouse in Puerto Rico.

Three different definitions are used to describe the results:

- A total number of candidates store locations: They are represented with plus signs in
- Figure 8.10, Figure 8.12, and Figure 8.13.
- Close stores cannot meet the system's demands and wishes regarding the constructs and objective function.

- Open stores are the stores that meet our requirements and satisfy our demand and wishes. They are represented with blue, red, and green in Figure 8.11 and 8.14 regarding their size.

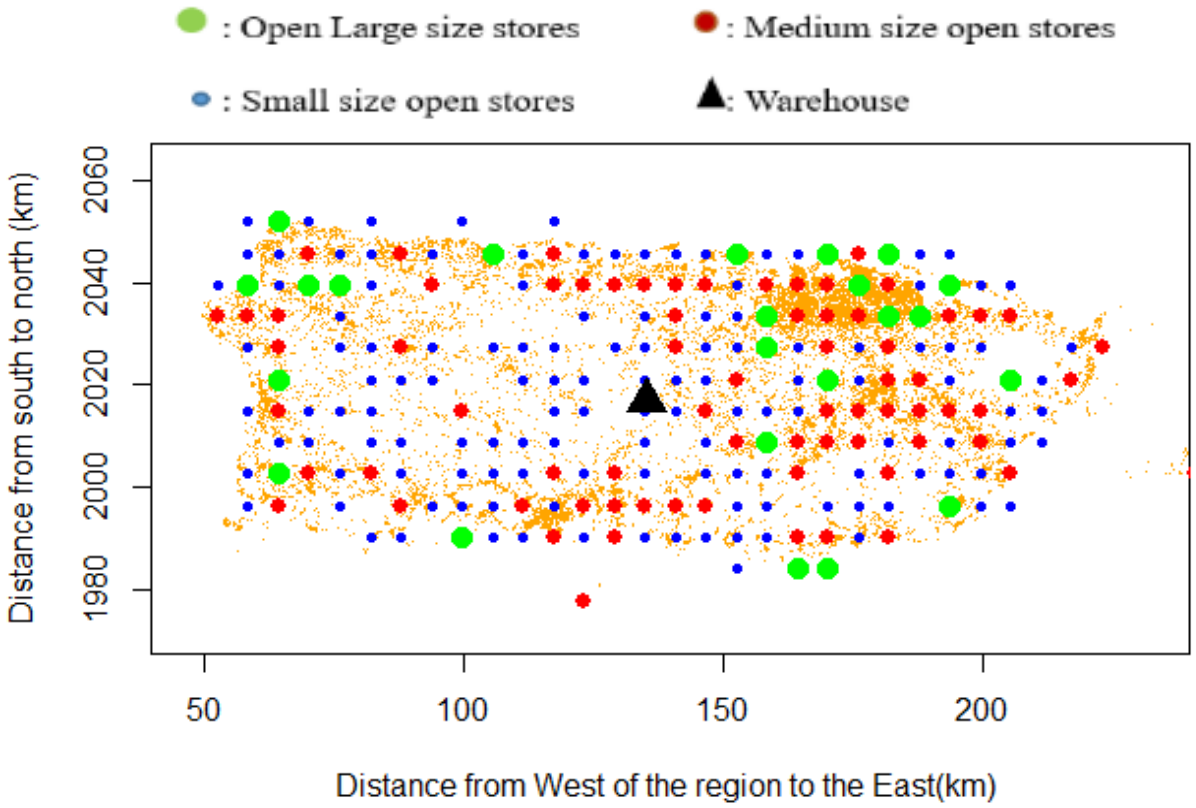


Figure 8.11. Open stores and warehouse in Puerto Rico.

The candidate store locations for this area are also presented on the same map (the plus signs). As can be seen, a uniform distance is employed between the candidate stores to simplify the model. Thus, in some cases, the stores are not exactly located inside the region due to this assumption. The model results for the whole island are shown in Figure 8.11. Three colors (blue, green, red) and sizes (small, medium, and large circles) are used to distinguish between the store sizes. As can be seen, the population and the warehouse distance are the main factors impacting the results. Thus, the large stores are mainly located near places with a high population, in this case near San Juan, the largest city on the map.

A few large stores can also be observed at locations near the map boundary with an average population and a large distance from the warehouse. Having large stores that are far from warehouse cause to reduce the truck's travel between these stores and warehouse to replenish the store more often and affects to reduce the truck's emission. Overall, the largest number of stores are small ones. Thus, for this case, it can be concluded that constructing small size stores is more beneficial considering the greenhouse gas emission as the deciding factor. More distributed small stores significantly reduce the travel distance of the customers to the store. As mentioned before, transportation plays an important role in increasing greenhouse gases.

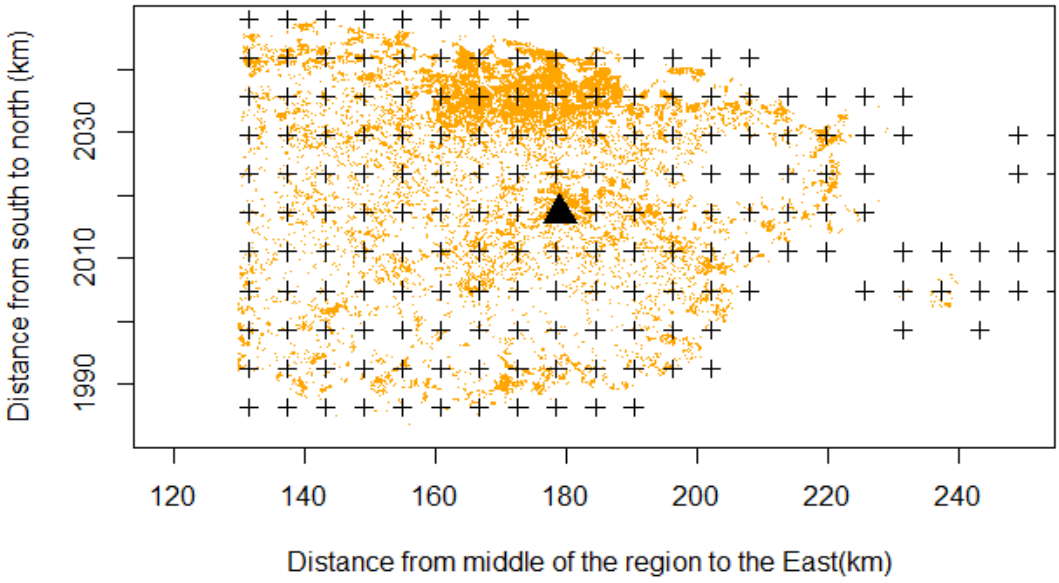


Figure 8.12. Candidate stores and warehouse on the east side of Puerto Rico.

The same analysis is repeated by dividing the region into two sections (east and west sides) and considering a separate warehouse in each region to show the sensitivity of the results to the location and number of warehouses. We show in Figure 8.12 and Figure 8.13 the computational domains considered for the region's east and west, respectively. The stores' location (the plus signs) and warehouse (the black triangles) are also shown in these figures.

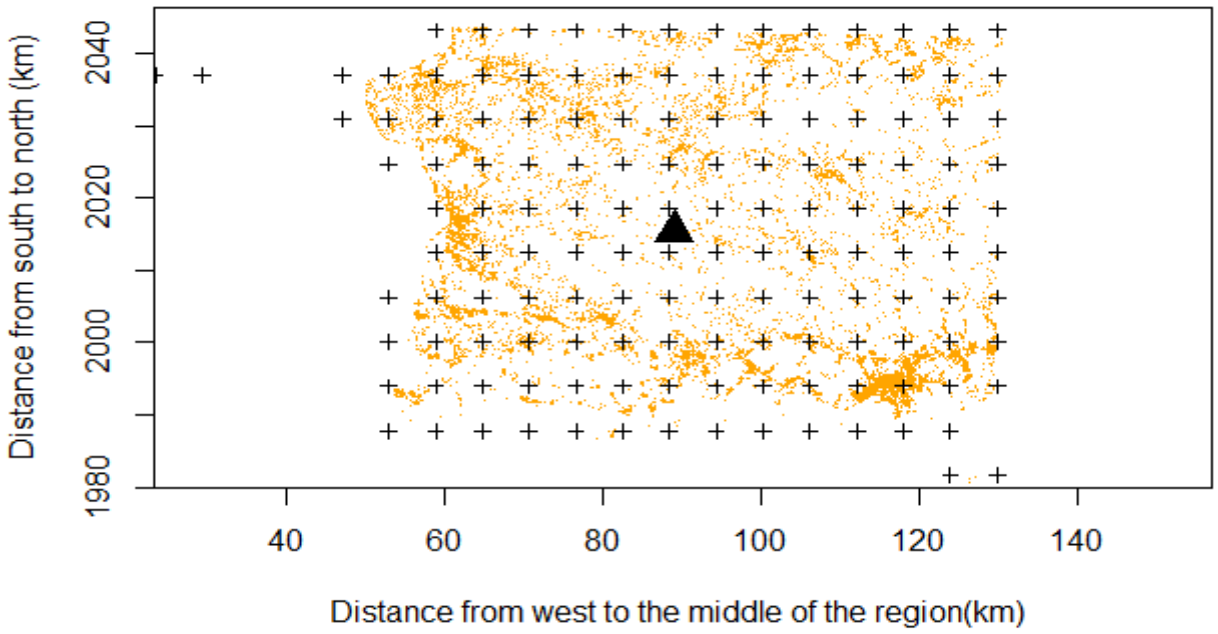


Figure 8.13. Candidate stores and warehouse in the west side of Puerto Rico

The total number of customer nodes on the east and west sides are 628 and 502, respectively. The total number of candidate stores in each region is 182 (east) and 149 (west). These numbers are slightly changed from the original case due to the fractals from the geographic information system (GIS) (Goodchild, 2011). There is some way to deal with this difficulty, like Fractal dimensions. This mathematical model expresses the idea that a line may be somewhere between one and two dimensional, with a fractal dimension of, say, 1.2 or 1.5 (fractal=fraction + dimensional). However, we avoid this complexity and accept small errors. In Figure 8.14, the case with two warehouses located on the east and west sides of Puerto Rico is shown. The arrangement of stores (locations and sizes) is different when two warehouses are considered in the whole region. For example, the total number of large stores on the west side is reduced to zero.

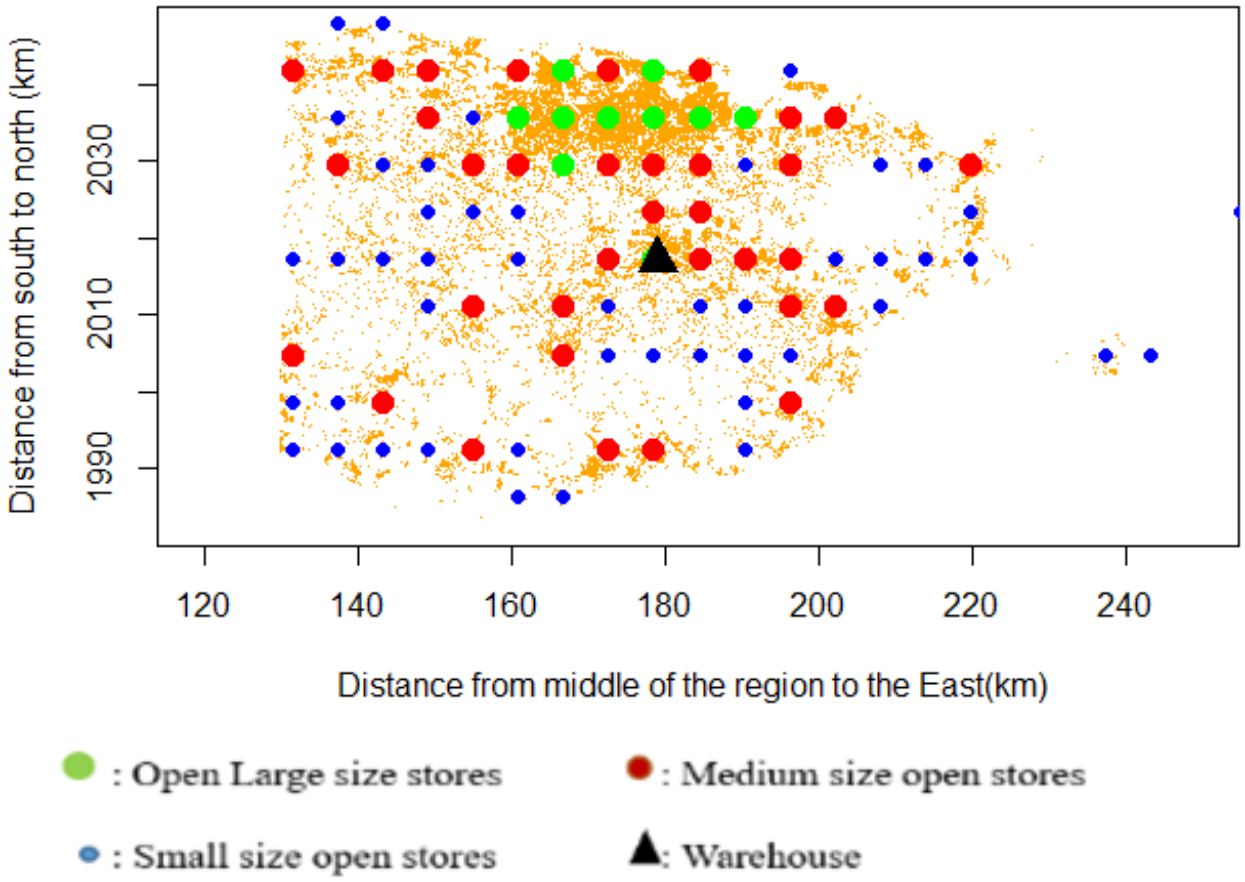


Figure 8.14. Open stores and warehouse on the east side of Puerto Rico.

As a general statement, the model tendency is toward reducing the number of large stores when the number of warehouses is increased. For scenario d, San Juan scenario, Puerto Rico's largest city, and capital, we consider 25 different customer nodes and 27 candidate stores (nine candidate location stores with three different store sizes, including large and small in each location). We have seven open stores, which include five large sizes and two small stores. Since San Juan is a crowded city, most of the open stores are large. The variations of results for the cases (b and c) considered earlier with one or two warehouses are shown in Figure 8.16 and Figure 8.17. The percentage of open stores with changing the number of warehouses has been shown in Figure 8.16, and the total number of open stores has been shown in Figure 8.17.

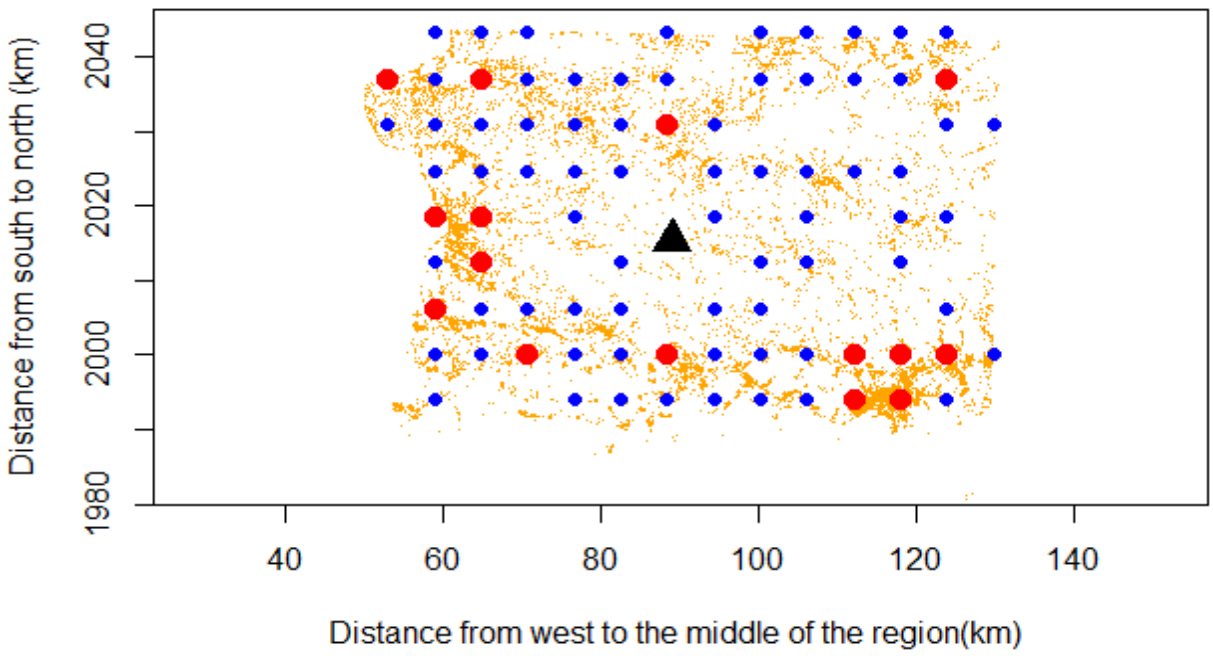


Figure 8.15. Open stores and warehouse on the west side of Puerto Rico.

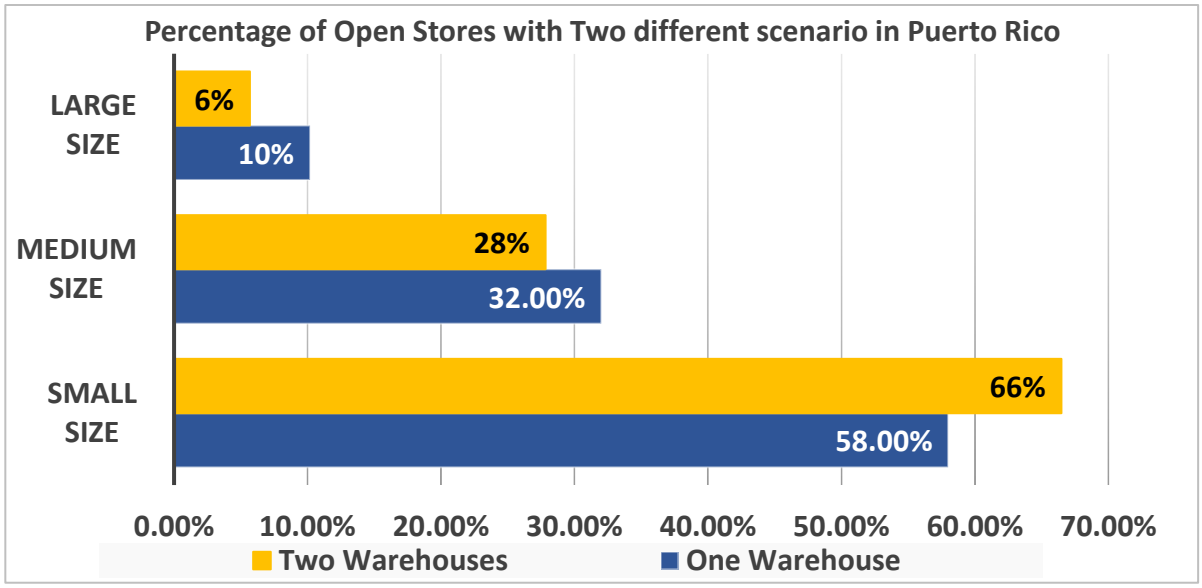


Figure 8.16. Percentage of open stores with changing the number of warehouses in Puerto Rico.

The following information can be derived from these figures. According to Figure 8.16, many small stores have fewer emissions than fewer, larger stores, confirming the base paper's results.

The model is sensitive to the number of warehouses considered in each region. When two warehouses are considered, fewer stores are open versus the previous results for one warehouse. This has been shown in Figure 8.17. As can be seen in Figure 8.16, with two warehouses, the majority of the open stores are small. Hence, regarding the result of having two warehouses in two different parts of the region, the operating cost and the emission cost of the system would be smaller (as we already considered the emission of buildings and warehouses in our system).

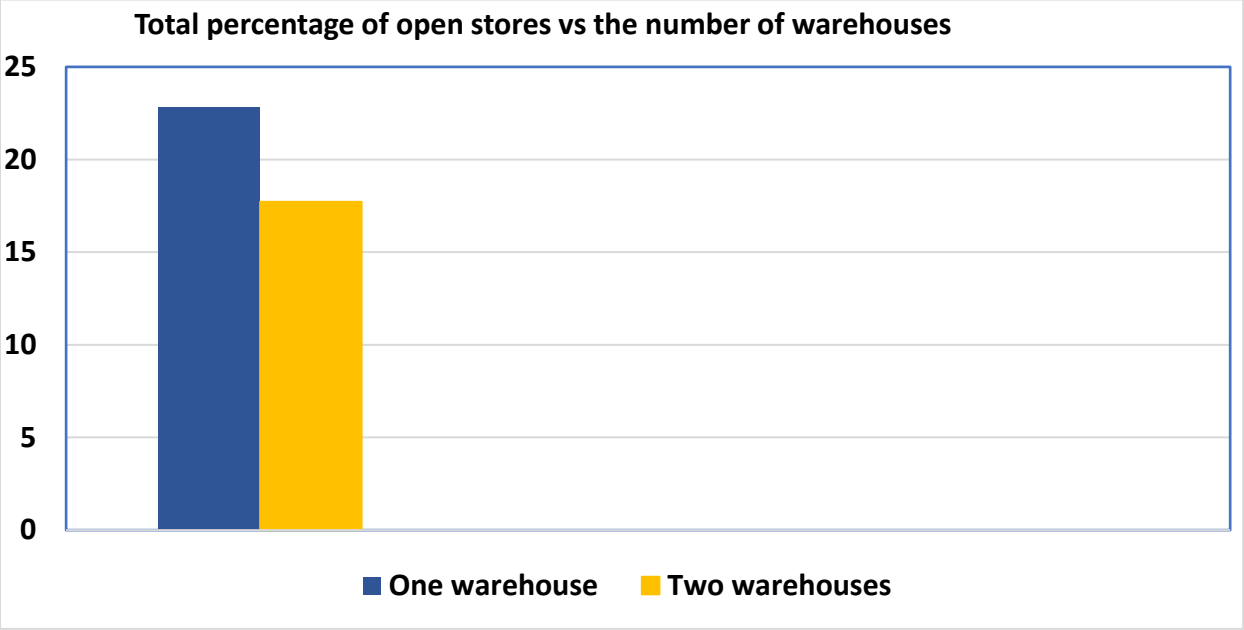


Figure 8.17. Total percentage of open stores in Puerto Rico with one warehouse vs. two warehouses.

When different weight factors are employed in the system goals (the desire to achieve a goal), the result is different. In

, we show the number and size of open stores on the east side of Puerto Rico when we desire to achieve the second system goal (operating cost-minimizing) shown in Equation 8.28.

We have fewer large size stores open when the goal is to minimize the operating cost. In Figure 8.15 for the east has represented the results when the desire is more toward the first system goal (emission minimizing) presented in Equation 8.27. We have more small size stores open when we consider emission minimization. As can be seen in this figure, the results of Williams et al. (Williams et al., 2020) and others (Cachon, 2014) are confirmed. Figure 8.18 concludea that the emission-minimizing system likely has more small stores in comparison with the operating cost-minimizing case, which has fewer large stores because operating cost increases from additional stores opening and trucks driving to them.

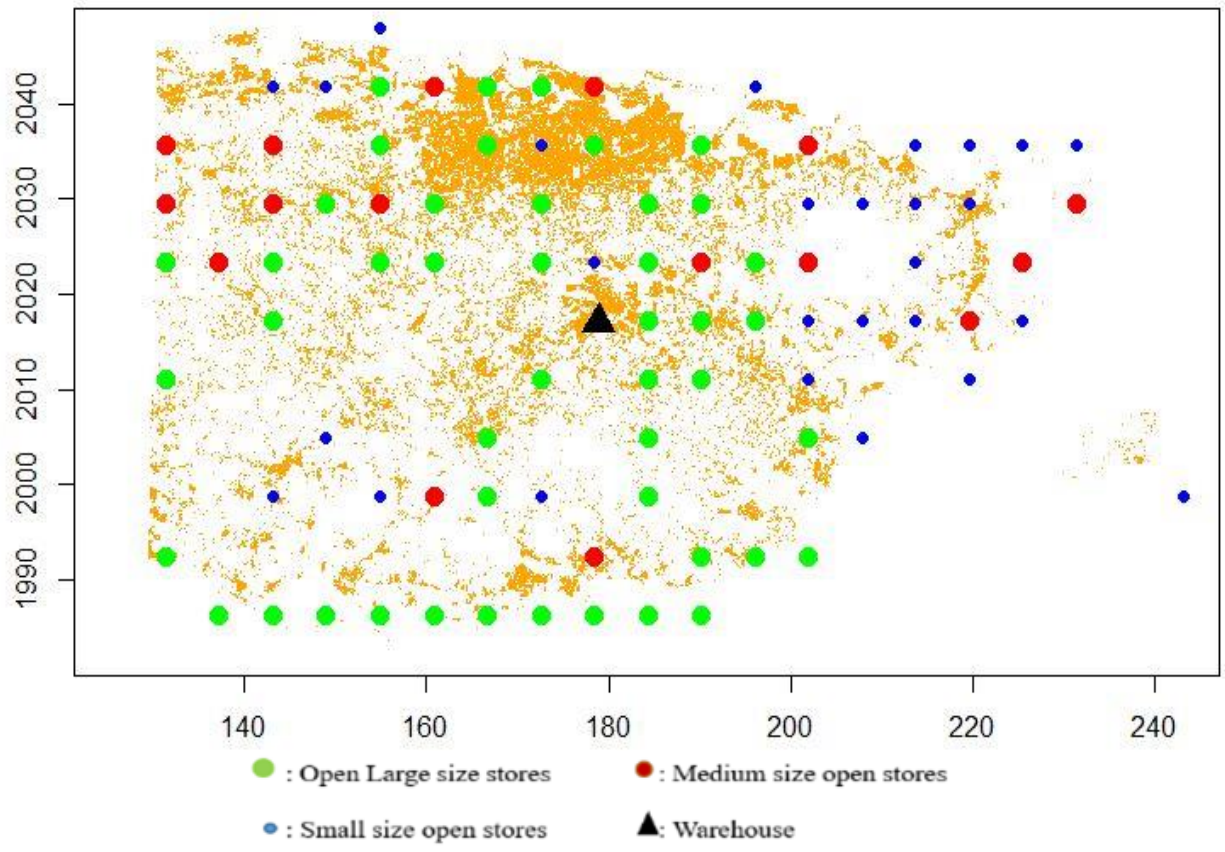


Figure 8.18. Open stores in the east side of Puerto Rico with considering more desire to the second system goal.

8.3.5. Verification and validation on the solution of asset management problem in a GSC

For verification and validation of the proposed model, the deviation variables are investigated to understand their limits. The number of goals for each scenario is shown in Table 8.7. While the number of goals in scenarios A, B, and C is a large number, the San Juan scenario is considered for this analysis as it has fewer deviation variables, and it is easier to see and describe the deviation parameters for this case. However, this method uses for all scenarios. For San Juan's scenario, we consider 25 different customer nodes and 27 candidate stores. Thus, there are 28 goals according to equations (4) and (5). Different weight factors for the system goals are defined (considering Equation (10)), and the behavior of deviation variables are investigated each time. Various considered weight factors are shown in Table 8.7.

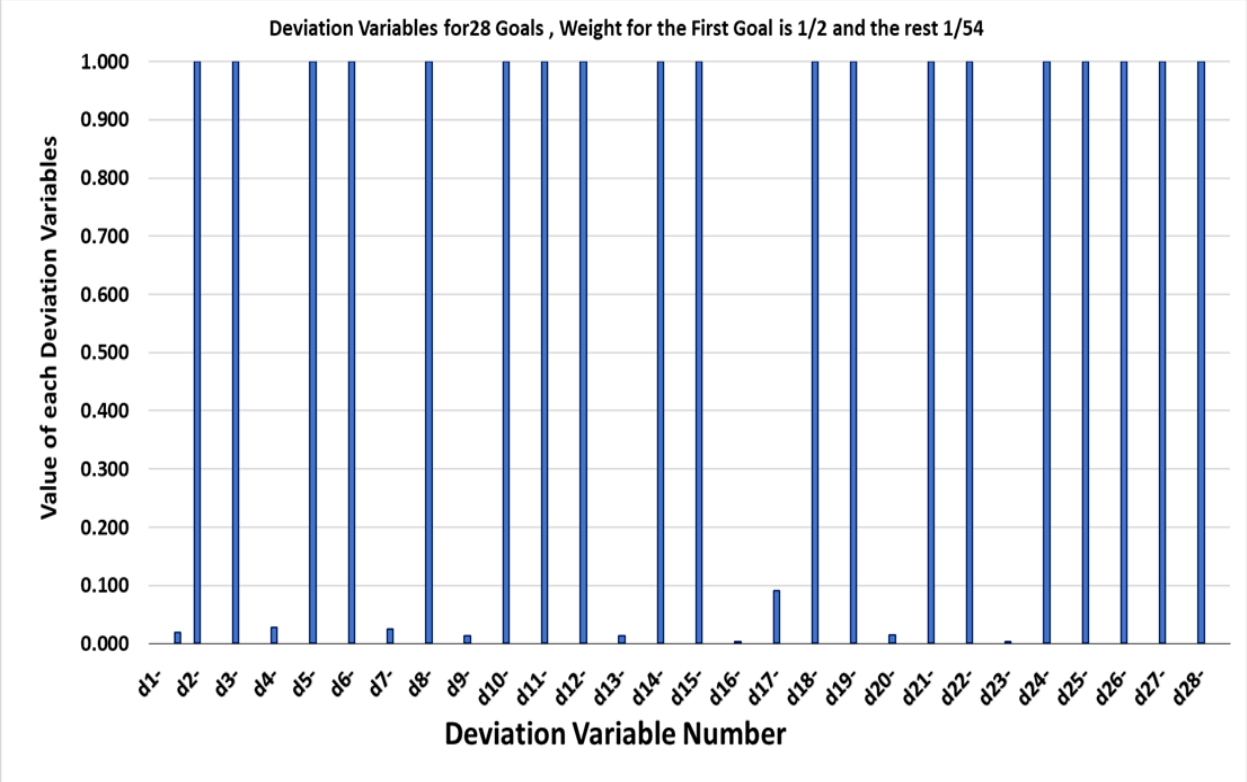


Figure 8.19. The value of deviation variables with W1=1/2 and other W=1/54.

Table 8.7. Different scenarios.

Scenario	Number of customer nodes	Number of candidate store	Number of goals
A (whole island)	1155	990	991
B (Westside of Puerto Rico)	502	447	448
C (Eastside of Puerto Rico)	628	546	547
D (San Juan)	25	27	28

Figure 8.19 for all goals are between zero and one for the cases considered. This confirms the program converges and the desired solution is determined.

Table 8.8. Different weight factors

Each scenario	Considered weight factors for first system goal (Equation 4)	Considered weight factors for the rest of the system goals (Equation 5)
a	1/28	1/28
b	1/4	1/36
c	1/2	1/54
d	2/3	1/81

As a result, we select scenario C, presented in

Figure 8.19. To build confidence in the results that are presented in this section, a convergence plot of the average deviation for the San Juan scenario is shown in Figure 8.13. We track the deviation of the goals at each iteration with the convergence plot.

As we see, the average deviations are reduced to about 6.6 E-4 after ~ 9 iterations. It means the program finds the correct solution, and no additional iterations are needed.

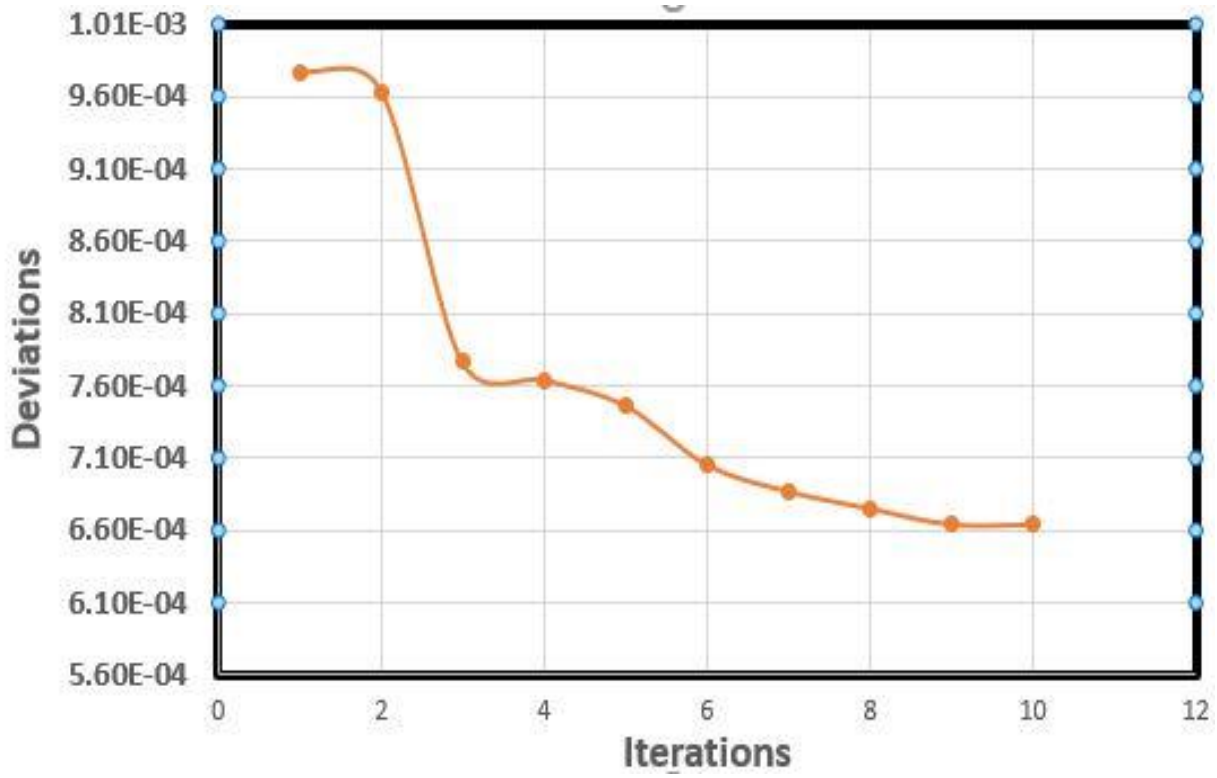


Figure 8.20. Deviation plotted against iteration.

8.3.6. Take aways of solving the asset management problem in a GSC

Global warming due to greenhouse emissions presents a severe threat to life on Earth, and the planet is warming to a degree beyond what many species can handle. Thus, reducing greenhouse gas emissions from different sectors should be considered for immediate action for government and private entities. According to an EPA (Environmental Protection Agency) report, three-quarters of greenhouse gas emissions from many industries are associated with their supply chains (EPA, 2010). Thus, reducing these gases from the supply chains is important for leading companies such as Walmart. These leading companies are also actively seeking ways to drive down emissions beyond their operations (EPA, 2010).

Green supply chain is valuable in sustainability for people(social), planet(environment) and profit (economic). In this section, we consider the effect of transportation on the green supply chain. In other words, this section is designed to help companies manage their resources to reduce their environmental footprint and understand influential decision factors and asset management's impact on GHG emissions reduction. It helps them to consider which kind of business model would be more effective.

Our model is developed based on the cDSP construct and is implemented in the MATLAB software. The model is employed to investigate the location and size of stores, considering population density. We consider three different scenarios to implement our model:

1) The whole island of Puerto Rico with one single warehouse (to consider the entire)
2) The city of San Juan (to consider the smaller area with a high population)
3) The whole island as two different parts, west and east parts, with one central warehouse in each region.

For reducing greenhouse gas emissions more smaller stores are more effective than fewer large stores. Hence, we need to have more smaller stores when we consider the cost of customers driving to the stores as car emissions are more influential to GHG emissions than the replenishment trucks and the emission from buildings. Thus, we justify that in the business model, under the standpoint of cost-efficiency in terms of labor, they have not considered the cost of customer driving to the big stores, which needs to contemplate in terms of GHG emissions.

Based on our results, this helps companies even consider changing their business model to the online business that eliminates the requirement of cars driving to the stores and back. In this case,

trucks can effectively deliver goods to large numbers of customers; thus, companies can reduce greenhouse gas emissions while reducing operating costs.

Considering the model presented here, different analyses can be performed to fully understand all the parameters impacting greenhouse gas emissions from a supply chain. One example is finding a single warehouse's best location regarding different factors such as population density, land availability, or distance to the main port, or considering multiple warehouses.

Opening multiple warehouses provide the opportunity to decrease the number of stores and connect the customers directly to the warehouses via online shopping channels. Subsequently, new business models evolve based on online shopping and its associated low carbon, pull economy. However, the number, size, and location of the warehouses need to be identified based on cost and emission factors to achieve the low-carbon configuration.

Moreover, there is an opportunity to investigate the challenges and differences between internet-based online shopping, which is a pull economy, and machine learning-based recommender systems, which cause push economy. A one-week operation limit is also implemented here that can be changed. For example, a customer can travel several times to a store in a week.

Finally, the proposed method here is not limited to GSC design. The concept can widely be extended to other domains such as manufacturing, material design, health care, and energy transmission and distribution. For example, material suppliers, healthcare providers, and energy distributors can be identified as cost-effective and generate less greenhouse gas emissions. Thus, further research along these lines looks promising.

8.4 Design A Bi-Level Programming Model for Two Channels in A Three-Echelon Supply Chain

We design a bi-level model where the upper level is to design the supply chain, including identifying the layout, the number of warehouses and stores, while the lower level is to identify the tour in online shopping. We use approximation (surrogate modeling) in the partitioning-coordination framework and making it partitioning-approximation-coordination framework, we propose an approach using surrogate approximation-based model where we manage the computational complexity by iteratively approximating the delivery tour function. This function is also called the lower-level function (the delivery tour for the delivery van in online shopping – Figure 8.21).

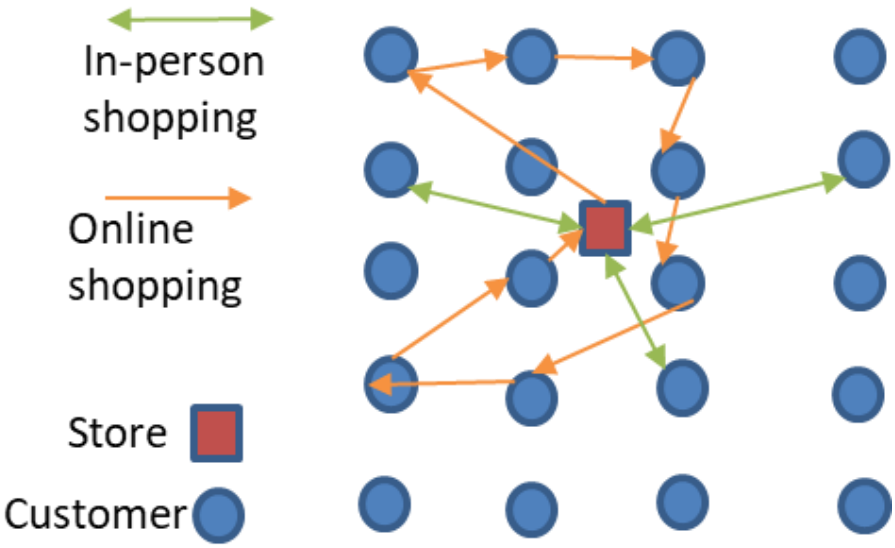


Figure 8.21. Difference between the distance customer travels in in-person shopping and the distance delivery van travels in online shopping (the delivery tour).

As shown in Figure 8.21, we propose a framework to design a multi-channel supply network, that includes in-store shopping or Brick and Mortar, and two types of online shopping, namely, Bricks and Clicks where customer orders online and picks up from the store and Pure Players

where only the retailer delivers. The proposed network is also multi-echelon as we model warehouses, stores, and customers. Also, because the demand is uncertain, we used spatial, socio-demographic, and economic factors to predict the demand as shown in the ZIGZAG lines in the picture. The resulting mixed-integer programming model is solved to minimize the cost and carbon emission together by monetizing carbon emissions (Greenness).

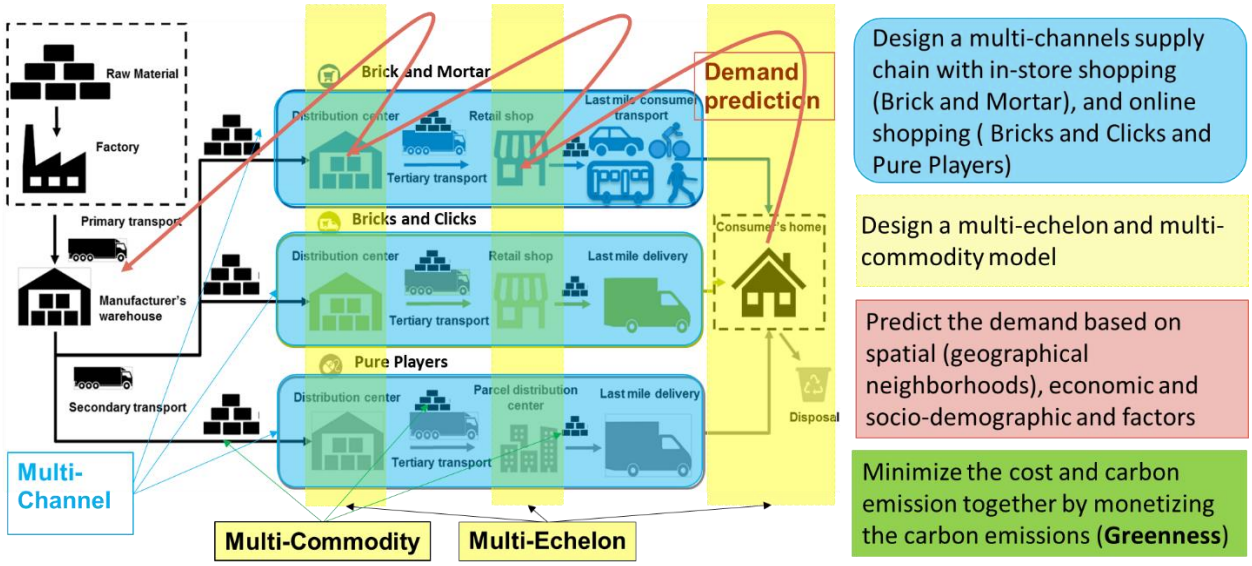


Figure 8.22. Framework for multi-echelon, multi-commodity, multi-channel, supply chain design with climate change mitigation perspective.

8.4.1. Mathematical model for multi-channel (online and traditional shopping), multi-echelon (warehouse, store, and customer) green supply chain

The mathematical model for multi-channel and multi-echelon green supply chain must have two parts: (1) the part which includes the design of the supply chain; (2) the part which includes identifying the **tour** in online shopping channel, which each delivery van (home delivery van which travels between stores and customers in online shopping channel) should travel in order to make all the deliveries.

1. **Part (1): supply chain design part**

$$\boldsymbol{min} \qquad \text{Equation 8.35}$$

$$\left(\sum_{w}\sum_{k} \boldsymbol{C1_k^w Z_k^w} + \sum_{s}\sum_{j} \boldsymbol{C2_j^s Y_j^s}\right) + \left(\sum_{j}\sum_{k} \boldsymbol{C3_k P_{kj}} + \sum_{j}\sum_{i} \boldsymbol{C4_j P_{ji}}\right) + \left(\sum_{k}\sum_{j} \boldsymbol{C5_{kj}} \frac{\boldsymbol{P_{kj}}}{\boldsymbol{CT}} (\boldsymbol{2d_{kj}})\right) + \boldsymbol{\alpha}\left(\sum_{j}\sum_{i} \boldsymbol{C6_{ji}} \frac{\boldsymbol{P_{ji}}}{\boldsymbol{CC}} (\boldsymbol{2d_{ji}})\right) + \boldsymbol{\beta}\left(\sum_{l\in(I\cup J)}\sum_{m\in(I\cup J)}\sum_{n\in N} \boldsymbol{C_{lm} R_{lmn} (d_{lm})}\right)$$

The objective function includes five parts. In the first part the opening cost is covered while the operating cost is covered in the second part. The third part includes transportation cost between the warehouse and store. The fourth part includes the transportation cost between store and customer in the traditional shopping while the transportation cost between store and customer in the online shopping is modeled in the fifth part.

Note that

$$C5_{kj} = C5_{kj}^{fm} + GI_1 * C_1 \; ; C6_{ji} = C6_{ji}^{fm} + GI_2 * C_2 \; ; C_{lm} = C_{lm}^{fm} + GI_3 * C_3$$

$C5_{kj}^{fm}, C6_{ji}^{fm}$ and C_{lm}^{fm} are the costs of fuel and maintenance of the retailer truck, customer car delivery van

C_1 , C_2 , *and* C_3 *are the costs of emission per* 1 *tone of* CO_2 *for truck* (*which delivers g*

$from\ a\ warehouse\ to\ a\ store$), *customer cars (in in-person shopping) and van (delivery van which delivers goods from stores to customers)*

GI_1 , GI_2 , *and* GI_3 *are the amounts of emission that truck, car and van creates* $(tone\ of\ CO_2)$

Let $G = (N, E)$ *be an undirected graph, where* $N = \{1, \dots n\} = I \cup J$ *is the set of nodes and E is he set of edges.* $C_{ij} = C_{ji}$ *for all* $i, j \in N$. *C satisfies the triangle inequality if and only if* $C_{lm} + C_{mh} \geq C_{lh}$ *for all l, m,* $h \in N$.

C_{ij} *is nonnegative and each customer has nonnegative demand.*

$$X_{ji}^{s} \leq Y_{j}^{s}, \qquad \forall j, i, s \qquad \text{Equation 8.36}$$

Store j is connected to customer i if store j with size s is open

$$\sum_{s} Y_{j}^{s} \leq 1, \qquad \forall j \qquad \text{Equation 8.37}$$

only one size of Store j is allowed to be open

$$V_{kj} \leq Z_{k}^{w}, \qquad \forall j, w \qquad \text{Equation 8.38}$$

Warehouse k is connected to Store j if Warehouse k with size w is open

$$\sum_{w} Z_{k}^{w} \leq 1 \qquad \text{Equation 8.39}$$

only one size of Warehouse k is allowed to be open

$$\sum_w \sum_s V_{kj}^{ws} \leq \sum_w Z_k^w$$ Equation 8.40

only one size of the store and one size of the warehouse is allowed to be open

$$\sum_w \sum_k V_{kj}^{ws} = Y_j^s, \quad \forall j, s$$ Equation 8.41

If Store j with size s is open, it must be connected to only one warehouse

$$P_{kj} \leq P_{kj}^{max} . \sum_w \sum_s V_{kj}^{ws}, \quad \forall k, j$$ Equation 8.42

flow capacity constraint to make sure that a flow of product from warehouse w to store j can take place only if the

$$P_{ji} \leq P_{ji}^{max} . \sum_s X_{ji}^s, \quad \forall j, i$$ Equation 8.43

flow capacity constraint to make sure that a flow of product from store s to customer zone i can take place only if the corresponding connection exists

$$P_{kj} \geq P_{kj}^{min} . \sum_w \sum_s V_{kj}^{ws}, \quad \forall k, j$$ Equation 8.44

minimum flow required for establishing the link k and j

$$P_{ji} \geq P_{ji}^{min} . \sum_s X_{ji}^s, \quad \forall i, j$$ Equation 8.45

minimum flow required for establishing the link I and j

$$\sum_k P_{kj} = \sum_i P_{ji}, \forall j$$ Equation 8.46

Flow balance constraint (inflow = outflow)

$$\sum_j P_{ji} = D_i, \qquad \forall i$$ Equation 8.47

Flow balance constraint (inflow = outflow), also demand constraint

$$\sum_s \sum_j X_{ji}^s = 1, \qquad \forall i$$ Equation 8.48

each customer can only connect to one store

$$\sum_k \sum_w \sum_s V_{kj}^{ws} = 1, \qquad \forall j$$ Equation 8.49

each store can only be supplied by one warehouse

$$\sum_j \sum_s D_j^s \sum_w V_{kj}^{ws} \leq \sum_w SP_k^w Z_k^w, \qquad \forall k$$ Equation 8.50

supply of the warehouse must be adequate to meet the demand of the stores (warehouse capacity constraint)

$$\sum_j \sum_s X_{ji}^s D_i PP_i \leq Y_j^s SP_j^s, \qquad \forall j$$ Equation 8.51

capacity of the store must be adequate to meet the demand of the customers (store capacity constraint)

$$Z_k^w, Y_j^s, V_{kj}, X_{ji} \in \{0,1\}, \qquad \forall j,s,k,i$$ Equation 8.52

$$P_{ji}, P_{kj} \geq 0, \qquad \forall j,k,i$$ Equation 8.53

all continuous variables must be non-negative

$$\sum_{n \in N} CV_n \, U_{jn}^s \;\leq\; SP_j^s Y_j^s, \qquad \forall j, s$$ Equation 8.54

The entire load for all the delivery vans operating on Store j must be less than the capacity of Store j

$$U_{jn}^s \;\leq\; Y_j^s, \qquad \forall j, s, n \in N$$ Equation 8.55

Van *n* can be assigned to a route originating from Store *j* of size S only if that store is open

$$\sum_s \sum_j U_{jn}^s \leq\; 1, \qquad \forall n$$ Equation 8.56

each van can be assigned to at most one store.

2. Part (2): Identifying the tour in online shopping

In this part we identify the d_{lmn}^{tour} after designing the supply chain, then, we solve the supply chain design part again with the new value for the d_{lmn}^{tour}. This is an iterative process and d_{lmn}^{tour} is the coupling variable between the two parts.

$$Min\left(\sum_{l \in (I \cup J)} \sum_{m \in (I \cup J)} \sum_{n \in N} C_{lm} d_{lm} \, R_{lmn}\right)$$ Equation 8.57

Lower-level function; minimizes the sum of the distance related routing costs

Subject to

$$\sum_{n\in N}\sum_{l\in(IUJ)} R_{lmn} = 1 \in (IUJ) \qquad \text{Equation 8.58}$$

every customer must be on exactly one route

$$\sum_{l\in(IUJ)}\sum_{m\in(IUJ)} R_{lmn} \;\leq\; CV_n \;\; \sum_{j\in J} U_{jn}^{s}\,, \qquad n \in N \qquad \text{Equation 8.59}$$

the capacity of delivery van for customer homes for the chosen route

$$\sum_{l\in(IUJ)} R_{lmn} \;= U_{mn}^{s}\; and \sum_{l\in(IUJ)} R_{mln}\,, \qquad \forall s,\, n,\, m \in (IUJ) \qquad \text{Equation 8.60}$$

flow consideration constraints:

if delivery van *n* is assigned to a route originating from store *m* of size *s*, then, at least one link goes into store *m*, and one levees store *m* (the van which enters to a node should leave that node)

$$\sum_{l\in st}\sum_{m\in st}\sum_{n\in N} R_{lmn} \leq |st| - 1, \qquad st = \{2,3,\dots,\,(i+j)\} \qquad \text{Equation 8.61}$$

Subtour elimination constraint

Parameters

$C1_k^w$ *fixed cost of opening warehouse k with size w*

$C2_j^s$ *fixed cost of opening store j with size s*

$C3_k$ *operational cost of handling product in warehouse k*

$C4_j$	*operational cost of handling product in store j*
$C5_{kj}$	*Transportation and emission cost of sending product from warehouse k to store j*
$C6_{ji}$	*Transportation and emission cost of customer i to buy product from store j*
C_{lm}	*Transportation and emission cost of delivering product from store j to customer i*
d_{lm}	*Distance between node l and node m (where l and m can be both stores and customers)*
P_{kj}^{max}	*maximum allowable flow on arc kj*
P_{ji}^{max}	*maximum allowable flow on arc ji*
P_{kj}^{min}	*minimum required flow on arc kj*
P_{ji}^{min}	*minimum required flow on arc ji*
D_i	*demand of customer i*
D_j^s	*demand of store j with size s*
SP_k^w	*supply capacity of warehouse k of size w*

SP_j^s	*supply capacity of store j of size s*
d_{kj}	*distance between warehouse k and store j*
d_{ji}	*distance between store j and customer i*
PP_i	*Population of customer zone i*
CT	*Capacity of truck resupplying stores from warehouses*
CC	*Capacity of customer cars*
$CV_{\boldsymbol{n}}$	*Capacity of delivery van* ***n*** *delivering to customers from stores*
α	*Share of in-person shopping market*
β	*Share of online shopping market*

Decision Variables

$Y_j^s = \begin{cases} 1 \\ 0 \end{cases}$	$store\ j\ size\ s\ is\ open$ $otherwise$
$Z_k^w = \begin{cases} 1 \\ 0 \end{cases}$	$warehouse\ k\ size\ w\ is\ open$ $otherwise$
$X_{ji}^s = \begin{cases} 1 \\ 0 \end{cases}$	$customer\ i\ is\ connected\ to\ store\ j$ $otherwise$

$V_{kj}^{ws} = \begin{cases} 1 \\ 0 \end{cases}$ $\quad$ $customer\ i\ is\ connected\ to\ store\ j$ / $otherwise$

$U_{jn}^{s} = \begin{cases} 1 \\ 0 \end{cases}$ $\quad$ $vehicle\ n\ operates\ for\ store\ j\ of\ size\ s$ / $otherwise$

$R_{lmn} = \begin{cases} 1 \\ 0 \end{cases}$ $\quad$ $Node\ l\ proceeds\ Node\ m\ on\ a\ route\ using\ Vehicle\ n$ / $otherwise$

P_{kj} Flow between k and j

P_{ji} Flow between j and i

The lower-level function is approximated by a surrogate model to reduce the two level task to one. My approach is to metamodeling the lower-level function and solving a number of auxiliary single level problems to obtain the solution for the bilevel problem. A feasible solution to the bilevel problem is a vector of upper and lower-level variables, such that, the vector satisfies all the constraints in the problem, and the lower-level variables (here the delivery tour distance) are the most appropriate values to the lower level problem for the given upper level variables (here all the distance and emission variables for both in-person and online shopping cases) as parameters.

8.4.2. Surrogate modeling to approximate the lower-level objective function

In this section, the idea for approximating the lower-level objective function is discussed. Considering the following bilevel problem in Definition 1, we define Lower-Level Value Function as Definition 2. Then we approximate the ϕ-mapping using Kriging and solve the reduced bilevel problem in Definition 2 using a standard single level algorithm. The process is carried out iteratively to converge toward the bilevel solution. This mapping relates the upper-level variables

with the corresponding objective function value of the lower-level problem. The method discussed in this section creates a meta-model of the value function mapping using Kriging approximations.

Table 8.9. Two definitions for the mapping function

Definition 1: for the upper level objective function $F: \mathbb{R}^n * \mathbb{R}^m \to \mathbb{R}$ and lower level objective function $f: \mathbb{R}^n * \mathbb{R}^m \to \mathbb{R}$, the bilevel problem is given by $\underset{x,y}{Min}\ F(x,y)$ Subject to $y \in argmin\{f(x,y): g_j(x,y) \leq 0, j = 1, \dots, J\}$ $G_k(x,y) \leq 0, k = 1, \dots, K,$ Where $G_k: \mathbb{R}^n * \mathbb{R}^m \to \mathbb{R}, k = 1, ..$ denotes the upper level constraints, and $g_k: \mathbb{R}$ $\mathbb{R}^m \to \mathbb{R}, j = 1, \dots, J$ represents the lower l constraints, respectively. Variables x and y a and m dimensional vectors, respectively.	Definition 2: let $\varphi: \mathbb{R}^n * \mathbb{R}^m \to \mathbb{R}$ be the lower level value function mapping, $\varphi(x) = \underset{y}{min}\{f(x,y): g_j(x,y) \leq 0, j = 1, \dots, J\}$ Represents the minimum lower level function value corresponding to any upper level decision vector. This is called as the lower level value function or the φ-mapping. The bilevel problem can be expressed as follows in terms of the φ-mapping: $\underset{x,y}{Min}\ F(x,y)$ Subject to $f(x,y) \leq \varphi(x)$ $g_j(x,y) \leq 0, j = 1, \dots, J$ $G_k(x,y) \leq 0, k = 1, \dots, K$

8.4.3. Surrogate-based approximation algorithm (partitioning – approximation - coordination framework)

Assuming the dimensions of the upper-level variables be **n** and the lower level be **m**.

1) generate random sample **S** of size **p = 10n** within the relaxed feasible region **Φ** of the bilevel problem. This is achieved by solving the following problem **p** times with random starting points. Let say that an algorithm **A** is used to solve the problem. Since the objective functions a fixed number, the algorithm terminates when a feasible solution is found.

$$\underset{x,y}{Min}\ 0 \qquad \text{Equation 8.62}$$

Subject to

$$G_k(x,y) \leq 0, k = 1, \dots, K \qquad \text{Equation 8.63}$$

$$g_j(x,y) \leq 0, j = 1, \dots, J \qquad \text{Equation 8.64}$$

$$x^{(i)}: i \in \{1, \dots, p\} \qquad \text{Equation 8.65}$$

2) solve the lower-level problem for each point in the sample $x^{(1)}, \dots, x^{(p)}$ using algorithm **A** and record the most appropriate point in the sample with the minimum upper-level function values $F^{(i)}\ and\ f^{(i)}: i \in \{1, \dots, p\}$. Initialize iteration counter $c \leftarrow c + 1$. Find the most appropriate point in the sample with the minimum upper-level function and denote it as F_c .

Now that the feasible solutions are acquired, the solution of the upper level's problem can be calculated. Therefore, we are now able to do the following:

- Find p samples from the feasible solution space of the relaxed problem
- Find the solution of lower level for each sample p and find the corresponding upper level objective value in the sample and the upper level objective value which is the first solution of the bi-level problem

To accomplish this step, we follow the following process:

- find the Euclidean distance between all customers and stores
- find the tour distance solution for each feasible solution from the shared feasible region (for both level problems)
- read the feasible solutions and select each store and its connected customers in each solution
- find the stores and customers' IDs for the current feasible solution
- build a list of coordination of nodes which are stores and customers that are connected in the current feasible solution
- calculate the Euclidean distance of connected stores and customers
- calculate the minimum tour distance for each set of connected store and customers
- calculate the sum of the tour distances for all the feasible solutions
- calculate F for each feasible solution (Decision 1 to Decision 32)

3) Compute the covariance matrix for the sample **S** and denote it by $\sum c$. Update $c \leftarrow c + 1$.

Output: $\sum c$. (The process of calculating the covariance matrix is explained in Appendices E and F)

4) estimate $\hat{\varphi}(x)$ (the approximated mapping from the sample **S** containing **p** points), and $\hat{s}(x)$ (the approximation of the standard deviation of $\varphi(x)$), and solve the following auxiliary problem to identify a new point x^{p+1} and it to sample population **S**.

$$\underset{x,y}{Min}\, F(x,y) \qquad \text{Equation 8.66}$$

Subject to

$$f(x,y) \leq \hat{\varphi}(x) + 3\hat{s}(x) \qquad \text{Equation 8.67}$$

Subject to

$$g_j(x,y) \leq 0, j = 1, \dots, J \qquad \text{Equation 8.68}$$

$$G_k(x,y) \leq 0, k = 1, \dots, K \qquad \text{Equation 8.69}$$

where **H** is the hypothesis space of functions, then one can identify an approximate model $\hat{\varphi} \in H$ that minimizes the empirical error on the sample (Equation 8.70); $L: \mathbb{R} * \mathbb{R} \rightarrow \mathbb{R}$ denotes the prediction error. The prediction error is commonly computed by assuming a quadratic loss function like follows:

$$\varphi\left(x^{(i)}\right) = \mu + \epsilon\left(x^{(i)}\right) \quad i = 1, \dots, n \qquad \text{Equation 8.70}$$

Where μ is the mean of the stochastic process and $\epsilon\left(x^{(i)}\right)$ is a normally distributed random error with mean 0 and variance σ^2.

5) Solve the lower level problem for this point using algorithm A to obtain f^{p+1} and compute F^{p+1}.

6) create additional random sample of n points (drawn from a normal random distribution with covariance 0.1*$\sum c$) with mean x^{p+1} and solve lower level problem for each point. Add these

additional sample points to $\boldsymbol{S}$ after computing the corresponding upper and lower level objective function values.

7) update $p \leftarrow p + 1 + n$.

8) for the current iteration find the point in the sample of $\boldsymbol{p}$ points with the minimum upper level function value and denote it as F_c .

9) terminate, if the improvement over multiple iteration is small, otherwise go to Step 3.

To accomplish Steps 3 and 4, we need to create a surrogate model to replace the mapping function (coordination function) between the lower level and upper level of the multi-channel supply chain. The regression model that we had created before in Figure 4 was not accurate and the reason for that is the nonlinear nature of the design points and the conceptual problem with regression which is the assumption of independent errors. This assumption is false when modeling a deterministic computer code, which is the case in our problem. Because the code is deterministic, any lack of fit is entirely modeling error (incomplete set of regression terms), not measurement error or noise. This means that the error terms are really collections of left-out terms in x, so that we may write $\epsilon^{(i)}$ as $\epsilon(x^{(i)})$. Moreover, if y(x) is continuous, then ϵ(x) is also continuous, because it is the difference between y(x) and the continuous regression terms. It follows that, if $x^{(i)}$ and $x^{(j)}$ are two points that are close together, then the errors $\epsilon(x^{(i)})$ and $\epsilon(x^{(j)})$ should also be close. In short, it makes no sense to assume that $\epsilon(x^{(i)})$ and $\epsilon(x^{(j)})$ are independent. Instead, it is more reasonable to assume that these error terms are related or 'correlated', and that this correlation is high when $x^{(i)}$ and $x^{(j)}$ are close and low when the points are far apart.

Therefore, instead of any regression model, we should use Kriging model. However, there is a problem with Kriging model which is not providing us with a function that we can replace the

lower level problem in the multi-channel supply chain design to approximate the lower level and turn it into a one level problem and solve it. To tackle this problem, we are proposing the following tasks to come up with a function for the Kriging model. This is also very helpful for design engineers as coming up with a function for the Kriging is a big challenge in engineering design.

As we do not assume that the errors are independent, but rather assume, as indicated above, that the correlation between errors is related to the distance between the corresponding points. We do not use the Euclidean distance, however, since this distance weights all the variables equally. Rather, we use the special weighted distance formula shown below:

$$\mathrm{d}(\mathrm{x}^{(\mathrm{i})}, \mathrm{x}^{(\mathrm{j})}) = \sum_{h=1}^{k} \theta_h \left|\mathrm{x}^{(\mathrm{i})} - \mathrm{x}^{(\mathrm{j})}\right|^{p_h} (\theta_h \geq 0,\ \mathrm{p}_h \in [1,2]). \qquad \text{Equation 8.71}$$

Using this distance function, the correlation between the errors at x(i) and x(j) is

$$\mathrm{Corr}[\epsilon(\mathrm{x}^{(\mathrm{i})}), \epsilon(\mathrm{x}^{(\mathrm{j})})] = \exp\left[-d(\mathrm{x}^{(\mathrm{i})}, \mathrm{x}^{(\mathrm{j})})\right]. \qquad \text{Equation 8.72}$$

The correlation function defined in (Equation 8.71) and (Equation 8.72) has all the intuitive properties one would like it to have. In particular, when the distance between x(i) and x(j) is small, the correlation is near one. Similarly, when the distance between the points is large, the correlation approaches zero. The parameter θ_h in the distance formula (Equation 8.71) can be interpreted as measuring the importance or 'activity' of the variable x_h. To see this, note that saying 'variable h is active' means that even small values of $|x_h^{(i)} - x_h^{(j)}|$ may lead to large differences in the function values at $x_h^{(i)}$ and $x_h^{(j)}$. Thinking in statistical terms, this means that even small values of $|x_h^{(i)} - x_h^{(j)}|$ should imply a low correlation between the errors ϵ(x(i)) and ϵ(x(j)). Looking at Equation 8.71

and Equation 8.72, we see that, if θ_h is very large, then it is indeed true that small values of $|x_h^{(i)} - x_h^{(j)}|$ translate to large 'distances' and hence low correlation. The exponent p_h is related to the smoothness of the function in coordinate direction h, with $p_h = 2$ corresponding to smooth functions and values near 1 corresponding to less smoothness.

It turns out that modeling the correlation in this way is so powerful that we can afford to dispense with the regression terms, replacing them with a simple constant term. This gives us the model we use in the stochastic process approach:

$$y(x(i)) = \mu + \epsilon(x(i)) \; (i = 1, \ldots, n) \qquad \text{Equation 8.73}$$

where μ is the mean of the stochastic process, $\epsilon(x^{(i)})$ is Normal $(0, \sigma^2)$, and, as just discussed, the correlation between errors is not zero but rather is given by Equation 8.71 and Equation 8.72. The estimates of the parameters μ and σ^2 have little direct interpretation, as they must be combined with the estimates of the correlation parameters (the θ_h's and p_h's) in order to make predictions.

This model is called a 'stochastic process model' because the error term $\epsilon(x)$ is a stochastic process, that is, it is a set of correlated random variables indexed by space (here, the k-dimensional space of x).

The model has 2k + 2 parameters: μ, σ^2, $\theta_1, \ldots, \theta_k$, and $p_1, \ldots, p_k$. We estimate these parameters by choosing them to maximize the likelihood of the sample. Let $y = (y^{(1)}, \ldots, y^{(n)})'$ denote the n-vector of observed function values, R denote the n×n matrix whose (i, j) entry is $\text{Corr}[\epsilon(x^{(i)}), \epsilon(x^{(j)})]$ (see eq. 2), and 1 denote an n-vector of ones. Then the likelihood function is:

$$\frac{1}{(2\pi)^{n/2}(\sigma^2)^{n/2}|R|^{1/2}} \exp\left[-\frac{(y-1\mu)'R^{-1}(y-1\mu)}{2\sigma^2}\right] \qquad \text{Equation 8.74}$$

The dependence on the parameters θ_h and p_h for h = 1,... ,k is via the correlation matrix R. Given the correlation parameters θ_h and p_h for h = 1,... ,k, we can solve for the values of μ and σ^2 that maximize the likelihood function in closed form:

$$\hat{\mu} = \frac{1'R^{-1}y}{1'R^{-1}1} \qquad \text{Equation 8.75}$$

And

$$\hat{\sigma}^2 = \frac{(y-1\hat{\mu})'R^{-1}(y-1\hat{\mu})}{n} \qquad \text{Equation 8.76}$$

Substituting Equation 8.75 and Equation 8.76 into the likelihood function, we get the so-called 'concentrated likelihood function', which depends only upon the parameters θ_h and p_h for h = 1,... ,k. This is the function that we maximize in practice to give us the estimates $\hat{\theta}_h$ and $\hat{p}_h$, and hence an estimate of the correlation matrix R. We then use Formulas Equation 8.75 and Equation 8.76 to get the estimates $\hat{\mu}$ and $\hat{\sigma}^2$. Formally, let r denote the n-vector of correlations between the error term at x^* and the error terms at the previously sampled points. That is, element i of r is $r_i(x^*)$ ≡ Corr[$\epsilon(x^*)$, $\epsilon(x^{(i)})$], computed using the formula for the correlation function in Equation 8.71 and Equation 8.72. It then turns out that the most appropriate linear unbiased predictor of y(x^*) is

$$\hat{y}(x^*) = \hat{\mu} + r'R^{-1}(y-1\hat{\mu}) \qquad \text{Equation 8.77}$$

On the right-hand side of Equation 8.77, the first term, $\hat{\mu}$, is the result of simply plugging x^* into the regression equation, and the second term represents the adjustment to this prediction based on the correlation of $\epsilon(x^*)$ with the error terms at the sampled points. Note that if there is no correlation (r = 0), then we just predict $\hat{y}(x^*) = \hat{\mu}$. We wanted to estimate the $\varphi(x)$ function which is the mapping function between the lower and upper level coordinating these two levels using maximum likelihood method as follows:

$$\hat{\varphi}(x) = \underset{u \in H}{argmin} \sum_{i \in I} L(x^{(i)}, f^{(i)}) \qquad \text{Equation 8.78}$$

On this statistical process, the maximum likelihood estimation is utilized to identify the Kriging parameters. The general form of the model is given as follows:

$$Z(x^{(i)}) = \mu + \epsilon(x^{(i)}) \quad \text{i} = 1, \dots, \text{n.} \qquad \text{Equation 8.79}$$

Here, $Z(x^{(i)})$ can be interpreted as a response obtained at $x^{(i)}$, μ is the term that represents the overall surface mean, and $\epsilon(x^{(i)})$ represents the autocorrelation error with 0 mean and σ^2 variance. The model assumes that the correlation between two points decrease as the distance between them increases. The correlation between any two points is given as follows:

$$C\left[\epsilon\left(x^{(i)}\right), \epsilon(x^{(j)})\right] = e^{-d(x^{(i)},\ x^{(j)})} \qquad \text{Equation 8.80}$$

Where $d(x^{(i)},\ x^{(j)} = \sum_{h=1}^{k} \theta_h \left|\text{x}^{(\text{i})} - \text{x}^{(\text{j})}\right|^{p_h}$ ($\theta_h \geq 0$, $\text{p}_h \in [1, 2]$). Represents the weighted distance between the two points $x^{(i)}$ and $x^{(j)}$). The parameters μ, σ, θ and p can be estimated by maximizing the likelihood over the sample.

Having the parameters estimated, the Kriging function can be estimated as follow:

$\hat{y}(x^*) = \hat{\mu} + r'R^{-1}(y - 1\hat{\mu})$

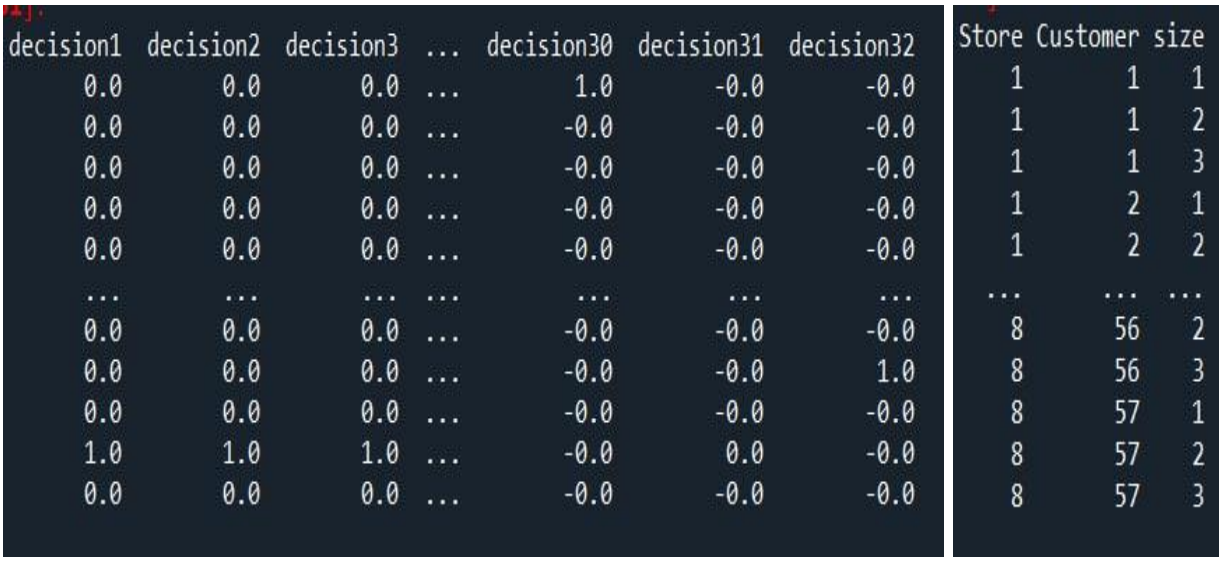
8.4.4. Results and discussion on solving the comprehensive problem of two channels in a three-echelon supply chain design problem

To demonstrate the Utility of the framework, we used Puerto Rico with 57 customer zones, and 7 candidate locations for stores (Total cost of the designed green supply chain is calculated considering \$100/ton CO_2).

1) generate random sample **S** of size **p = 10n** within the relaxed feasible region **Φ** of the bilevel problem.

We wrote the Python code for the bi-level program where the upper level is the supply chain design model, and the lower level is the tour distance determination model for the delivery van in the online shopping. Then, we identified the feasible solution space of the bi-level problem by relaxing the objective functions of the upper and lower level and solving the relaxed problem for *p* times. We found p feasible solutions with random starting points as sample *S*. The first sample of the feasible solutions for the bi-level problem is shown in Figure 8.23.

We show here which store is connected to which open store. Please note that we just wanted to show a schematic outcome and we could not plot the whole outcome here as this is not the final result and we plot the final results in proper figures and visualizations. $x^{(1)}, \dots, x^{(p)}$ using algorithm **A** and record the most appropriate point in the sample with the minimum upper-level function values $F^{(i)}\ and\ f^{(i)}: i \in \{1, \dots, p\}$. Initialize iteration counter $c \leftarrow c + 1$. Find the most appropriate point in the sample with the minimum upper-level function and denote it as F_c .



decision1	decision2	decision3	...	decision30	decision31	decision32
0.0	0.0	0.0	...	1.0	-0.0	-0.0
0.0	0.0	0.0	...	-0.0	-0.0	-0.0
0.0	0.0	0.0	...	-0.0	-0.0	-0.0
0.0	0.0	0.0	...	-0.0	-0.0	-0.0
0.0	0.0	0.0	...	-0.0	-0.0	-0.0
...	...	...	...	...	...	...
0.0	0.0	0.0	...	-0.0	-0.0	-0.0
0.0	0.0	0.0	...	-0.0	-0.0	1.0
0.0	0.0	0.0	...	-0.0	-0.0	-0.0
1.0	1.0	1.0	...	-0.0	0.0	-0.0
0.0	0.0	0.0	...	-0.0	-0.0	-0.0

Store	Customer	size
1	1	1
1	1	2
1	1	3
1	2	1
1	2	2
...	...	...
8	56	2
8	56	3
8	57	1
8	57	2
8	57	3

Figure 8.23. Feasible solution for the bi-level problem

For each solution from *S*, we solved the lower-level problem, which is the tour finding problem and identified the solution for each sample points in the sample pool of *S*. In Figure 8.23, we show the tours and corresponding values for the sample points. Please note that each decision in the figure stands for all the open stores in that decision (design) with its corresponding customers and accordingly several tours in each decision (design). Again, since this is not the final results, we just took a screen shot from the Python output to show several tours and their corresponding costs. Also, the sequence of the customers is show in the figure which in the final results are visualized on Puerto Rico's map with customers sequence in the tours.

In Figure 8.24, we show the final tour for each decision (design), where the values are the total tour distance in each design. Also, we plotted a mapping for the first sample (32 datapoints or designs). As you see the mapping is not explaining the tour identification function appropriately. Therefore, we try other mappings and iteratively improve the quality of the mapping.

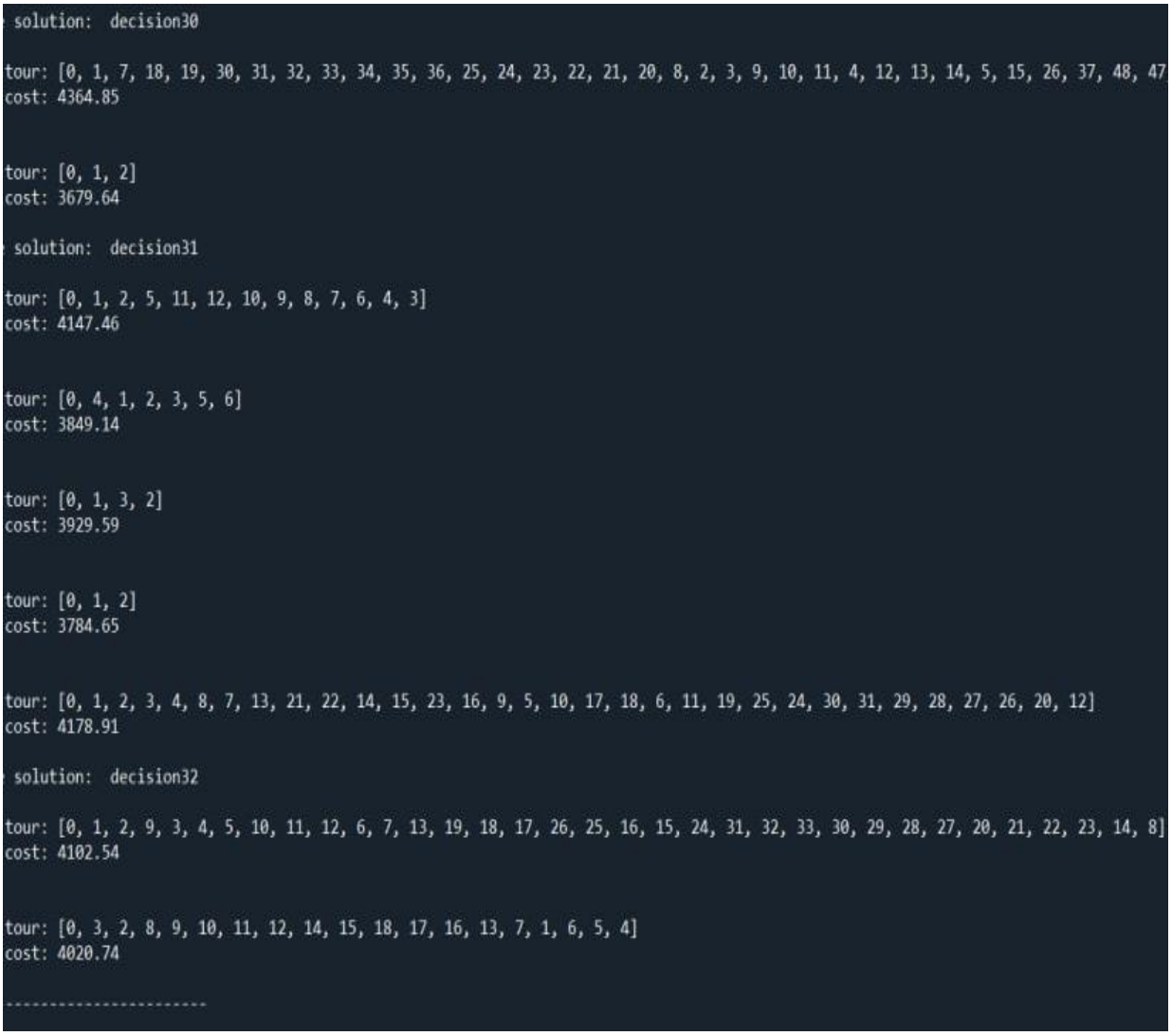


Figure 8.24. Tour distance in online shopping for each sample point and the corresponding cost of each tour.

These solutions are feasible for the upper level, and we solve the upper level in coordination with the lower-level solutions as the next step. This is the solution of the bi-level problem (traditional, in-store shopping and online shopping together) for the first sample set of points and the first iteration. We wrote a code to automate this process and explore the solution space and add new sample points to the pool of samples.

```
In [725]: Final_tour
Out[725]:
{'decision1': 31266.22693879271,
 'decision2': 31266.22693879271,
 'decision3': 31266.22693879271,
 'decision4': 31266.22693879271,
 'decision5': 31131.023456660943,
 'decision6': 31131.023456660943,
 'decision7': 31208.002695583942,
 'decision8': 31208.002695583942,
 'decision9': 31266.22693879271,
 'decision10': 31266.22693879271,
 'decision11': 31728.92032718494,
 'decision12': 31266.22693879271,
 'decision13': 30731.275329676522,
 'decision14': 30485.478901297665,
 'decision15': 30664.430140246535,
 'decision16': 30887.315646649346,
 'decision17': 31333.514508356748,
 'decision18': 31333.514508356748,
 'decision19': 31003.202185681923,
 'decision20': 31003.202185681923,
 'decision21': 31125.790623251192,
 'decision22': 31125.790623251192,
 'decision23': 31090.699633941007,
 'decision24': 31090.699633941007,
 'decision25': 30360.188859671136,
 'decision26': 30360.188859671136,
 'decision27': 31227.15982524721,
 'decision28': 30397.196999794498,
 'decision29': 30886.511930755878,
 'decision30': 30894.268478980855,
 'decision31': 30970.96722774954,
 'decision32': 30458.384100868865}
```

decision1	31266.23
decision2	31266.23
decision3	31266.23
decision4	31266.23
decision5	31131.02
decision6	31131.02
decision7	31208
decision8	31208
decision9	31266.23
decision10	31266.23
decision11	31728.92
decision12	31266.23
decision13	30731.28
decision14	30485.48
decision15	30664.43
decision16	30887.32
decision17	31333.51
decision18	31333.51
decision19	31003.2
decision20	31003.2
decision21	31125.79
decision22	31125.79
decision23	31090.7
decision24	31090.7
decision25	30360.19
decision26	30360.19
decision27	31227.16
decision28	30397.2
decision29	30886.51
decision30	30894.27
decision31	30970.97
decision32	30458.38

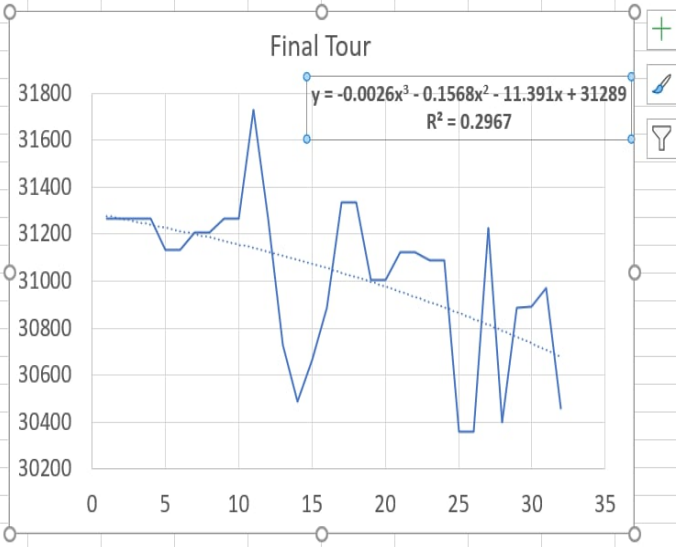


Figure 8.25. Final tour for each decision (design) and mapping for the first sample (32 datapoints or designs).

Now that the feasible solutions are acquired, the solution of the upper level's problem can be calculated. Therefore, we are now able to do the following:

Find p samples from the feasible solution space of the relaxed problem

Find the solution of lower level for each sample p and find the corresponding upper-level objective value in the sample and the upper-level objective value which is the first solution of the bi-level problem. The results are shown in Table 8.10.

Table 8.10. Solution for the upper-level function

0	1	2	3	4	5	6	7	8	9	10
0.0	9688.7	9688.7	9688.7	9688.7	9688.7	9688.7	9688.7	9688.7	9688.7	9688.7
11	12	13	14	15	16	17	18	19	20	21
9688.7	9688.7	9688.7	9688.7	9688.7	9688.7	16216.6	16216.6	3741.1	3741.1	4918.1
22	23	24	25	26	27	28	29	30	31	32
4914.4	4469.6	4469.6	12864.6	12864.6	4787.3	4787.3	4612.3	4612.3	5039.3	5039.3

2) Calculating covariance matrix of the 32 random solutions (each denoted by d1 (solution1) and d2 (solution 2))

```
('31', '10'): 1.0003000450045003 ('31', '26'): 7.016748962402968,
('31', '11'): 1.0003000450045003 ('31', '27'): 1.0,
('31', '12'): 1.0003000450045003 ('31', '28'): 1.0,
('31', '13'): 1.0003000450045003 ('31', '29'): 1.0001000050001667,
('31', '14'): 1.0003000450045003 ('31', '30'): 1.0001000050001667,
('31', '15'): 1.0003000450045003 ('31', '31'): 1.0,
('31', '16'): 1.0003000450045003 ('31', '32'): 1.0,
('31', '17'): 15.678651344022036 ('32', '1'): 1.0010005001667084,
('31', '18'): 15.678651344022036 ('32', '2'): 1.0003000450045003,
('31', '19'): 1.0005001250208359 ('32', '3'): 1.0003000450045003,
('31', '20'): 1.0005001250208359 ('32', '4'): 1.0003000450045003,
('31', '21'): 1.0003000450045003 ('32', '5'): 1.0003000450045003,
('31', '22'): 1.0003000450045003 ('32', '6'): 1.0003000450045003,
('31', '23'): 1.0001000050001667 ('32', '7'): 1.0003000450045003,
('31', '24'): 1.0001000050001667 ('32', '8'): 1.0003000450045003,
('31', '25'): 7.016748962402968, ...}
```

Figure 8.26. Results of calculating covariance matrix of the 32 random solutions.

3) Formulating variance and mean value and build the likelihood function

$$\hat{\mu} = \frac{1'R^{-1}y}{1'R^{-1}1} \qquad \text{Equation 8.81}$$

$$\hat{\sigma}^2 = \frac{(y - 1\hat{\mu})'R^{-1}(y - 1\hat{\mu})}{n} \qquad \text{Equation 8.82}$$

In Figure 8.27, we show the module that we created to calculate the covariance matrix, the inverse of the covariance matrix, $\hat{\mu}$ and $\hat{\sigma}^2$. Calculating these components enables me to estimate the parameters of the likelihood function. This allows me to create a function for the Kriging method.

$$\frac{1}{(2\pi)^{n/2}(\sigma^2)^{n/2}|R|^{1/2}} \exp\left[-\frac{(y - 1\mu)'R^{-1}(y - 1\mu)}{2\sigma^2}\right] \qquad \text{Equation 8.83}$$

```
L={}
Cov={}
for d1 in solutions:
    for d2 in solutions:
        L[d1,d2]= 0.0001 * round( (
              Tetha1*sum ( pow( float (X_ijs_f[j,i,s,d1]) - float(X_ijs_f[j,i,s,d2]), Pi1) for j in J for i in I for s in S )
            + Tetha2*sum (pow( float(Y_js_f[j,s,d1]) -  float(Y_js_f[j,s,d2]) , Pi2)  for j in J for s in S)
            + Tetha3*sum (pow( float(V_kwjs_f[k,j,w,s,d1]) -  float(V_kwjs_f[k,j,w,s,d2]) , Pi3)  for k in K for w in W for j in J for s
            + Tetha4*sum (pow( float(Z_kw_f[k,w,d1]) -  float(Z_kw_f[k,w,d2]) , Pi4)  for k in K for w in W)
            + Tetha5*sum (pow( float(P_ji_f[j,i,d1]) -  float(P_ji_f[j,i,d2]) , Pi5)  for j in J for i in I)
            + Tetha6*sum (pow( float(P_kj_f[k,j,d1]) -  float(P_kj_f[k,j,d2]) , Pi6)  for k in K for j in J)
            + Tetha7*sum (pow( float(R_lmn_f[l,m,n,d1]) -  float(R_lmn_f[l,m,n,d2]) , Pi7 ) for l in M for m in M for n in N)
            + Tetha8*sum (pow( float(U_jns_f[j,n,s,d1]) -  float(U_jns_f[j,n,s,d2]) , Pi8)  for j in J for n in N for s in S) )  , 6)

        Cov[d1,d2]= math.exp (-L[d1,d2])

# turn Cov dict to a matrix
import numpy as np
Cov_mat = np.zeros((len(solutions), len(solutions)))
for d in Cov:
    i = solutions.index(d[0])
    j = solutions.index(d[1])
    Cov_mat[i, j] = Cov[d]+ 0.00000001* randint(0,10)


# compute the inverse of cov
Inv_Cov_mat=np.linalg.inv(Cov_mat)

# convert Inv_Cov_mat to a dictionary

Inv_Cov={}
for i in range (len(Inv_Cov_mat)):
     for j in range(len(Inv_Cov_mat[i])):
            Inv_Cov[i+1,j+1]=Inv_Cov_mat[i][j]


# formulation for Mu
Mu= sum (f[j] * sum (Inv_Cov[i,j] for i in range (1,33)) for j in range (1,32)) / sum (Inv_Cov[i,j]
                                                    for i in range (1,33) for j in range (1,32))
# formulation for sigma2
sigma2= (1/32) * sum ( pow ( (f[i]-Mu), 2) * Inv_Cov[i,j] for i in range (1,33) for j in range (1,32) )
sigm= math.sqrt(sigma2)
```

Figure 8.27. The module for calculating the $\hat{\mu}$ and $\hat{\sigma}^2$ and the likelihood function.

In Figure 8.28, we show the results of calculating the likelihood function parameters.

```
In [75]: print(Tetha1, Tetha3,Tetha4,Tetha8,Tetha5,Tetha6,Tetha7,Tetha8,Tetha2)
0.1 0.1 0.15 0.13 0.08 0.08 0.08 0.13 0.13

In [76]: print(Pi1,Pi2,Pi3,Pi4,Pi5,Pi6,Pi7,Pi8)
1 1 1 1 1 1 1 1
```

Figure 8.28. Results of calculating the likelihood function parameters.

Now, that we have the likelihood function, we are able to formulate the bi-level model into a single level model by the following formulation:

$$\underset{x,y}{Min}\, F(x,y)$$

Subject to

$$F(x,y) \leq \hat{\varphi}(x) + 3\,\hat{s}(x) \qquad \text{Equation 8.84}$$

$$G_k(x,y) \leq 0, k = 1, \dots, K \qquad \text{Equation 8.85}$$

$$g_j(x,y) \leq 0, j = 1, \dots, J \qquad \text{Equation 8.86}$$

Where $\hat{s}(x)$ is calculated as following:

4) Estimating the likelihood function parameters and the mean and variance as well as the s^2:

$$S^2(X^*) = \sigma^2 \left[1 - r'R^{-1}r + \frac{(1 - 1'R^{-1}r)^2}{1'R^{-1}1}\right] \qquad \text{Equation 8.87}$$

The result of calculating the $s^2(x)$ is shown in Figure 8.29.

```
In [75]: print(Tetha1, Tetha3,Tetha4,Tetha8,Tetha5,Tetha6,Tetha7,Tetha8,Tetha2)
0.1 0.1 0.15 0.13 0.08 0.08 0.08 0.13 0.13

In [76]: print(Pi1,Pi2,Pi3,Pi4,Pi5,Pi6,Pi7,Pi8)
1 1 1 1 1 1 1 1
```

Figure 8.29. Results of calculating S^2(x).

Now, the single level model results are achievable as shown in Figure 8.30.

```
In [68]: X_value, Y_value, Z_value, V_value, P_kj_value, P_ji_value,R_lmn_value,U_jns_value, OBJ_value= Krig (1.26, 1.23, 1.26  ,
Mu, sigm,sigma2,Inv_Cov, solutions, f, X_ijs_f,Y_js_f,Z_kw_f,V_kwjs_f,P_kj_f,P_ji_f,R_lmn_f,U_jns_f)
Parameter OutputFlag unchanged
   Value: 1  Min: 0  Max: 1  Default: 1
Gurobi Optimizer version 9.1.1 build v9.1.1rc0 (win64)
Thread count: 2 physical cores, 4 logical processors, using up to 4 threads
Optimize a model with 3915 rows, 37414 columns and 280892 nonzeros
Model fingerprint: 0xb3a8dd65
Variable types: 512 continuous, 36902 integer (36902 binary)
Coefficient statistics:
  Matrix range     [1e+00, 3e+08]
  Objective range  [1e-02, 5e+03]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+05]
Warning: Model contains large matrix coefficients
         Consider reformulating model or setting NumericFocus parameter
         to avoid numerical issues.
Presolve removed 59 rows and 1040 columns
Presolve time: 0.37s
Presolved: 3856 rows, 36374 columns, 118020 nonzeros
Variable types: 480 continuous, 35894 integer (35894 binary)

Root relaxation: objective 1.250365e+04, 2938 iterations, 0.14 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

*    0     0               0    12503.652146 12503.6521  0.00%     -    0s

Explored 0 nodes (2938 simplex iterations) in 0.70 seconds
Thread count was 4 (of 4 available processors)

Solution count 1: 12503.7

Optimal solution found (tolerance 1.00e-04)
Best objective 1.250365214610e+04, best bound 1.250365214610e+04, gap 0.0000%
computational time= 0.7341372966766357
```

Figure 8.30. Results for the bi-level model which is transferred to a single level model.

6) create additional random sample of n points (drawn from a normal random distribution with covariance 0.1*$\sum c$) with mean x^{p+1} and solve lower-level problem for each point. Add these additional sample points to $\boldsymbol{S}$ after computing the corresponding upper and lower-level objective function values.

Now, we have the $\hat{\varphi}(x)$ function and we can add a new sample point to the pool of samples and calculate its corresponding solution and update the most appropriate lower and upper level solutions.

Choose another 32 samples within 0.1 covariance and run the all the previous steps to find the most promising lower and upper level solution. This process is continued until the values of the lower and upper levels do not change. The current set of solutions, including 32 random sample solutions and the solution obtained by the Kriging model built based on the random sample solution are shown in Figure 8.31. The lowest value of the upper level functions belongs to the Solution 19 with the value of F= 3741.08 as it is shown in Figure 8.32.

Value of f(x,y), lower level, tour finding problem

```
34: 2240.4475395798554,
35: 1213.0319527886338,
36: 2218.1149619688745,
37: 1317.1797008546528,
38: 1278.3720066600288,
39: 2121.883134191149,
40: 2114.490601311075,
41: 2320.0345001272863,
42: 2386.1822586361036,
43: 1248.2899868460224,
44: 2294.1197479163634,
45: 1288.1797090097189,
46: 1138.2868358861194,
47: 2263.6931782330626,
48: 1290.445135995439,
49: 1275.0082297276922,
50: 2255.252353413517,
51: 1217.5108545441076,
52: 1224.437239566142,
53: 2287.303956181879,
54: 2345.8503633535856,
55: 1180.9958276853395,
56: 1239.2772204261444,
57: 1196.1739986839582,
58: 1398.163086380066,
59: 1208.536115391554,
60: 2153.6857853682423,
61: 1144.9781738470022,
62: 1375.7984597003044,
63: 1139.969676122053,
64: 2172.4448753885913,
65: 2139.6984208313092}
```

Value of F(x,y), upper level, supply chain design problem

```
34: 9393.859726368104,
35: 8266.444139576883,
36: 9079.920889916475,
37: 8630.591887642902,
38: 8428.784193448279,
39: 11116.648199468615,
40: 9964.902788099324,
41: 10995.799565404752,
42: 14408.514111192015,
43: 8422.782066454223,
44: 11455.488643424218,
45: 8722.67178861792,
46: 12957.618688442031,
47: 13900.025030788975,
48: 13159.77698855135,
49: 11763.970900780569,
50: 4947.510953223899,
51: 3573.655439770599,
52: 5879.913745237988,
53: 4764.562555992261,
54: 4965.108963163968,
55: 5828.8462103175625,
56: 8953.160389934246,
57: 5104.6505043558045,
58: 2795.4601140186555,
59: 2657.46618129171,
60: 4170.151323894011,
61: 3804.0790905638323,
62: 2127.0954873388937,
63: 8427.726630555217,
64: 8424.065869239159,
65: 8376.319414681877}
```

```
In [888]: F_best= min(F.items(), key=lambda x: x[1])

In [889]: F_best
Out[889]: (62, 2127.0954873388937)
```

Figure 8.31. Results for the lower level and upper-level objective function using the first set of sample solutions and the Kriging solution.

```
In [756]: F_best= min(F.items(), key=lambda x: x[1])

In [757]: F_best
Out[757]: (19, 3741.078529727729)
```

Figure 8.32. The lowest value of upper-level objective function among the first 33 solutions.

7) update $p \leftarrow p + 1 + n$.



Figure 8.33. The results for the second 32 samples for the tour finding (value of *f(x,y)*, the lower-level problem).

The change in the lowest solution for upper level (*F*) is significant and equal to: F(13) – F(62)= 3741.08-2127.09= 1613.99 which indicates an almost 40% reduction. We still need to keep taking the samples from normal distribution with mean of the current lowest solution and 0.1*covariance of the current 62 solution because the change in the problem solution *F* is higher than 5% which we have set as the stopping criteria.

Value of f(x,y), lower level, tour finding problem

Value of F(x,y), upper level, supply chain design problem

```
In [888]: F_best= min(F.items(), key=lambda x: x[1])

In [889]: F_best
Out[889]: (62, 2127.0954873388937)
```

```
34: 2240.4475395798554,
35: 1213.0319527886338,
36: 2218.1149619688745,
37: 1317.1797008546528,
38: 1278.3720066600288,
39: 2121.883134191149,
40: 2114.490601311075,
41: 2320.0345001272863,
42: 2386.1822586361036,
43: 1248.2899868460224,
44: 2294.1197479163634,
45: 1288.1797090097189,
46: 1138.2868358861194,
47: 2263.6931782330626,
48: 1290.445135995439,
49: 1275.0082297276922,
50: 2255.252353413517,
51: 1217.5108545441076,
52: 1224.437239566142,
53: 2287.303956181879,
54: 2345.8503633535856,
55: 1180.9958276853395,
56: 1239.2772204261444,
57: 1196.1739986839582,
58: 1398.163086380066,
59: 1208.536115391554,
60: 2153.6857853682423,
61: 1144.9781738470022,
62: 1375.7984597003044,
63: 1139.969676122053,
64: 2172.4448753885913,
65: 2139.6984208313092}
```

```
34: 9393.859726368104,
35: 8266.444139576883,
36: 9079.920889916475,
37: 8630.591887642902,
38: 8428.784193448279,
39: 11116.648199468615,
40: 9964.902788099324,
41: 10995.799565404752,
42: 14408.514111192015,
43: 8422.782066454223,
44: 11455.488643424218,
45: 8722.67178861792,
46: 12957.618688442031,
47: 13900.025030788975,
48: 13159.77698855135,
49: 11763.970900780569,
50: 4947.510953223899,
51: 3573.655439770599,
52: 5879.913745237988,
53: 4764.562555992261,
54: 4965.108963163968,
55: 5828.8462103175625,
56: 8953.160389934246,
57: 5104.6505043558045,
58: 2795.4601140186555,
59: 2657.46618129171,
60: 4170.151323894011,
61: 3804.0790905638323,
62: 2127.0954873388937,
63: 8427.726630555217,
64: 8424.065869239159,
65: 8376.319414681877}
```

Figure 8.34.Value of f(x,y), the lower level prolem and F(x,y), the upper level prolem and the updated lowest solution for upper level (F) for the second 32 samples.

The change in the lowest solution for upper level (*F*) is not significant and equal to: F(62)-F(92)= 2127.09-2069.36= 57.73 which is about to 2% change in the previous solution. Since we considered a threshold of 5% change, we stop here.

8.4.5. Scenario analysis on different percentages of online shoppers in a two-channel, three-echelon supply chain

Now that the solutions for the bi-level program can be achieved, we can conduct different scenario analysis based on different percentages of the in-store and online shoppers for calculating

the difference in expected carbon emissions between in-store shopping and e-commerce-based online retailing involving last mile delivery to customers' homes to quantify which channel has the least harmful impact on the environment. The goal here is to answer the questions of "Is online shopping greener than the traditional shopping? Under which conditions (uncertain)?" and we hypothesize that online shopping is greener if the delivery truck travels a short distance. For example, as shown in Figure 15, if the percentage of online shoppers increases the overall GHG emission of the supply chain is expected to be increased.

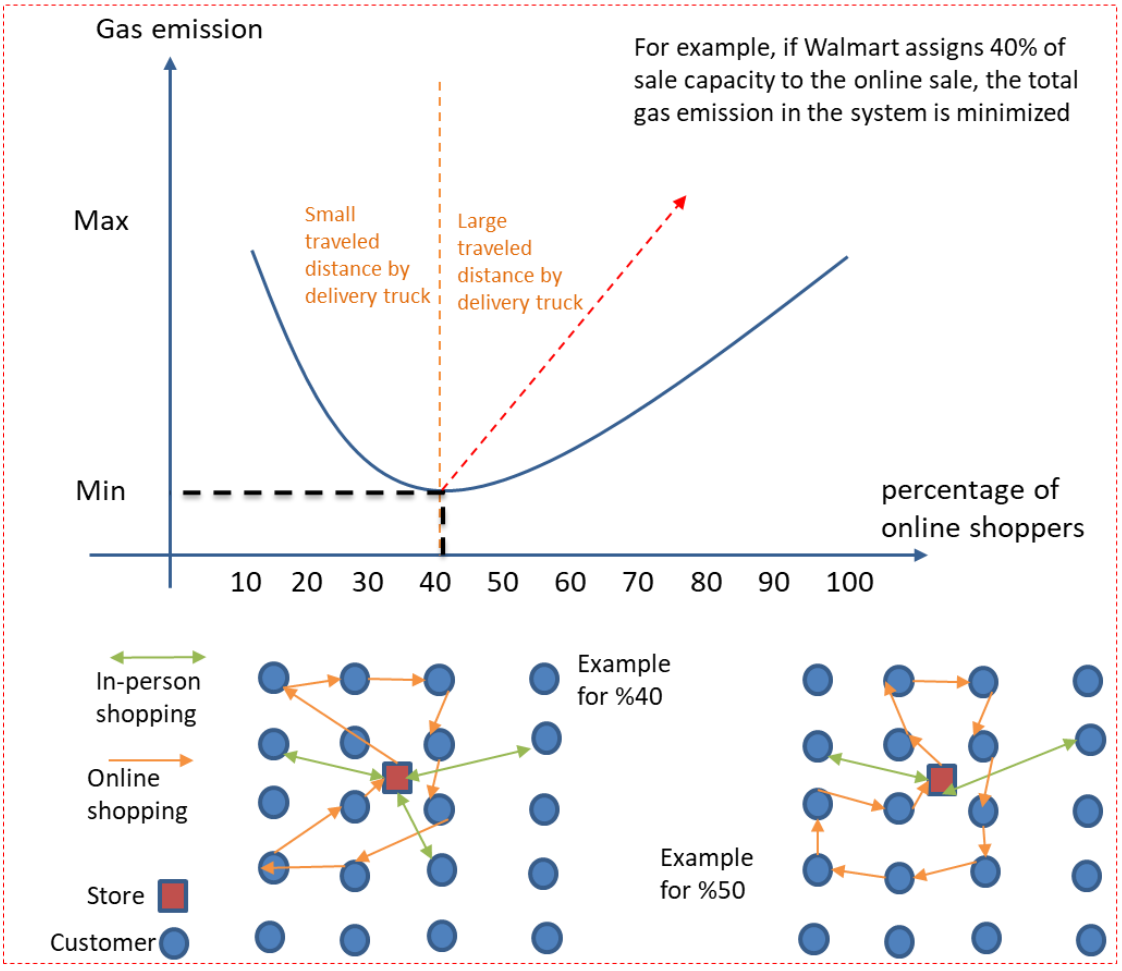
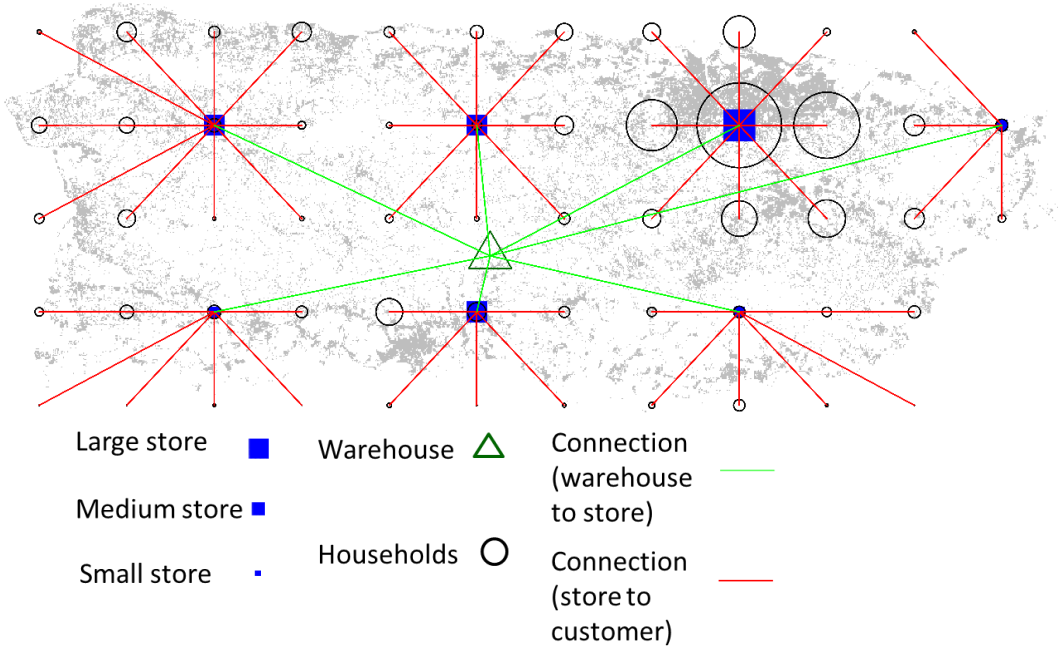


Figure 8.35. the hypothetical different between different percentages of online shoppers.

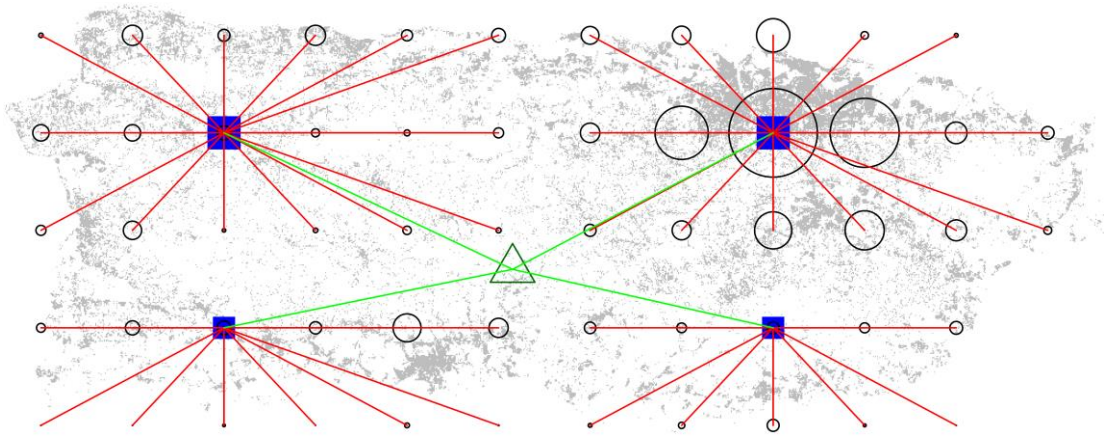
Scenario 1: $\boldsymbol{\alpha}$ = 0.8, $\boldsymbol{\beta}$ = 0.2, more in-store shopping (*Assumption: 80% of transportation cost is related to customers who buy in store. In fact, 80% of the customers buy in store or in other words, percentage of online shoppers is 80%.*)

Total Cost = 6.54e+04



Scenario 2: $\boldsymbol{\alpha}$ = 0.2, $\boldsymbol{\beta}$ = 0.8, more online shopping

Total Cost = 84.62e+04



Scenario 3: $\boldsymbol{\alpha}$ = 0.55, $\boldsymbol{\beta}$ = 0.45, a tad more traditional shopping

Total Cost = 5.87e+04

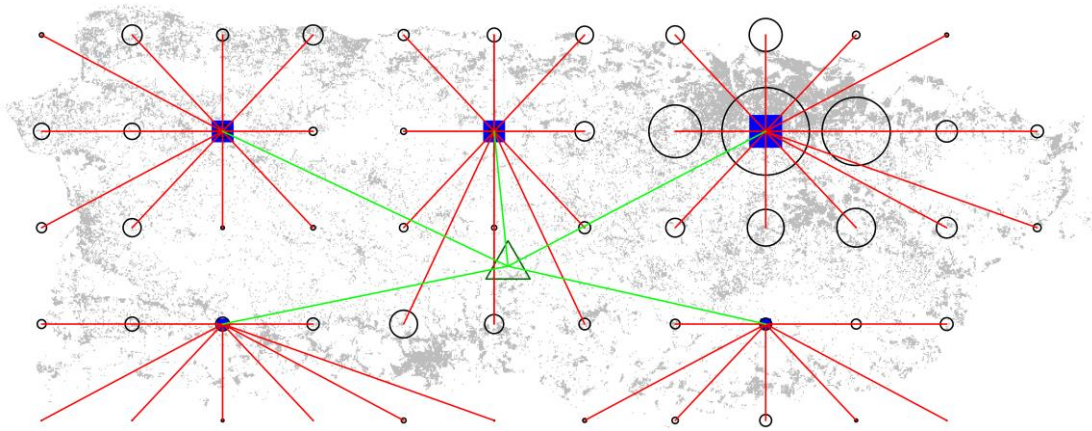


Figure 8.36. Difference scenarios based on different percentages of the online shoppers.



Figure 8.37. Total cost of the supply chain based on percentage of online shoppers.

Online shopping does not appear to be cleaner than in-store shopping, so my hypothesis is not proven to be right. As the percentage of the online shoppers increases, the total cost including the greenhouse gasses emissions decreases. However, as soon as it reaches around 40%-50%, the cost of the system design increases. This might be mainly due to the increase in customer travel distance.

8.4.6. Verification and validation of solving the comprehensive problem of two channels in a three-echelon supply chain design problem

Theoretical structural validation

Theoretical structural validation refers to accepting the validity of two portions of the dissertation, including the (i) surrogate modeling and (ii) partitioning-coordination and accepting the internal consistency of the way these two portions are put together for partitioning-approximation-coordination framework. Theoretical structural validation involves systematically identifying the scope of the framework's application, reviewing relevant literature and identifying the research gaps that is existing, identifying the strengths and limitations of the surrogate-based mathematical modeling used based on literature review, determining the approaches that can be leveraged for developing the partitioning-approximation-coordination framework, reviewing literature on the advantages, disadvantages and accepted domains of application, and checking the internal consistency of the constructs (Portions i and ii of the dissertation) both individually and when integrated into the framework.

The internal consistency of the individual constructs is checked by a critical review of the literature. The verification and validation of Research Question 6, which is "What is the mathematics to partition, approximate and coordinate a complex multi-channel and multi-echelon supply chain design problem?" is carried out in detail in Sections 8.14 to 8.19. The readers are referred to this section for more details. To answer this question, we hypothesized that using a bi-

level partitioning approach, we can model two different channels (online and in-store shopping), then we can use a surrogate model to map the solutions of the lower-level into the upper-level. The key outcome that we obtained by doing this research project and completing Chapter 8 is creating a method for calculating the difference in expected carbon emissions between in-store shopping versus e-commerce-based online retailing involving last mile delivery to customers' homes to quantify which channel has the least harmful impact on the environment.

8.4.7. Take aways on solving the comprehensive problem of two channels in a three-echelon supply chain design problem

In this chapter we determine the sizes and facility locations of stores in an area of customers to reduce overall supply chain cost and greenhouse gas emission simultaneously such that total supplier's restocking expenses, the supplier's space expenses, and the customers' transportation expenses are minimized. These expenses include variable operating fees, such as gas and rent and electricity for supplier spaces. These expenses also represent the fuel and electricity cost as indirect expenses related to GHG emissions, but it is possible that, in this model, the full prices of GHGs are not included.

Store restocking is carried out by trucks going back and forth from the central warehouse to each store instead of requiring the trucks to use a traveling salesman pattern. Using the traveling salesman, trucks which empty their loads in the delivery to the first store are still required to travel to all stores and produce unnecessary emissions. This we argue, is not as realistic as a model in which trucks travel back and forth between specific stores and the warehouse. In this chapter, truck resupply is modelled with a k-median problem and is superimposed on the customer distribution which is also a k-median problem; thus, sustainable supply chain store distribution becomes a bi-level problem. Further, instead of assuming that demand is uniform in the region

under consideration, we base demand on the population distribution, using the examples of Puerto Rico and its capital city, San Juan.

Two contrasting scenarios are considered for supply chain density design: (1) minimizing operating costs without regard to GHG emissions; and (2) GHG emission minimization, regardless of operating charges. Some cases are insensitive to which scenario is chosen; when the charges for supplier space are low, or if the electricity used by the supplier is produced with high GHGs or if the supplier's gas consumption is high, GHG emissions are near minimum if operating costs are minimized, and operating costs are near minimum although emissions are minimized. This occurs when all the sources of emissions have similar GHG emissions relative to operating charges. For example, in our model, although supplier trucks can carry a lot more product than customer cars, the GHGs for every unit of distance to variable operating costs for every unit of distance is approximately equal for cars and trucks.

There are also circumstances when the ratios of the GHG emissions to the operating charges are different. If the supplier is in a region with more expensive rental rates and consumes comparatively environmentally friendlier electricity, the ratio of the associated GHGs to operating charges for floor space varies substantially from the ratio for cars (perhaps as much as 0.13 kilograms carbon dioxide per dollar versus 1.36 kilograms carbon dioxide per dollar). In this situation, a supply chain which minimizes emission costs has a condensed network of tiny stores to obtain the advantage of the supplier's environmentally friendly electricity compared to the customers' pollutant vehicles, while a supply chain which minimizes operating costs has a sparse network to obtain the advantage of the inventory effectiveness of big stores. A supply chain which reduces GHGs is denser than one which reduces operating charges for about 7-fold. Moreover,

minimizing operating charges rises GHGs by approximately 70%, whereas minimizing the GHGs causes an equal penalty for the operating charges.

Using the bi-level program created for the multi-echelon, multi-channel green supply chain design in a push-pull economy, we enable retailers like Walmart to:

- Adapt post Covid-19 market and build green supply chains
- Design a new business model transitioning from a push economy to an on-line based pull economy.
- Predict the online and in-store demand
- Show which retail channels: traditional or online shopping is greener and under which conditions.
- identify the number, size, and location of the stores and warehouses in a supply chain to achieve a green and low-cost configuration.

This proposed design method is not limited to GSC design. It is easily extended to other supply chain designs including those in manufacturing, material design, healthcare, and energy transmission and distribution.

CHAPTER 9 CLOSING REMARKS, I STATEMENT AND WAY FORWARD, I STATEMENT

In Chapter 9, the dissertation is summarized, and the intellectual contributions are critically reviewed. The advantages and limitations of the methods, metrics, and constructs are discussed. For theoretical performance validation, it is argued that these constructs are valid beyond the example problems selected for empirical validation. Finally, avenues for future research and broader applications of the fundamental ideas in this dissertation are discussed.

9.1 Problem and Assumptions

In this dissertation, the problem that have been dealing with is

- To manage complexity using surrogate modeling (SM), which are predictive models and partitioning
- To deal with feature-based, time-based, and spatial data in SM
- To partition complex design problems to smaller parts and then put them together

After a critical evaluation of the literature, several gaps are identified in the literature and described as follows:

- There is a gap in SM pool of knowledge in creating ensemble of SM when the data is limited
- There is a gap in automating the SM selection process without removing the human intelligence
- Temporal and spatial SM needs to be studied more
- Partitioning, approximation, and coordination can be integrated for complexity management without losing the nature of the problem

To address these gaps, two main technical approaches are utilized as surrogate modeling and partitioning. In terms of surrogate modeling, ensemble of SM as weighted combinational surrogate models, multi-layer SM selection approach to automatically choosing the appropriate surrogate model, time series analysis, and spatial statistics are used. Moreover, in terms of partitioning, k-means partitioning, bi-level programming, and surrogate-based approximation as solution approach are used.

In order to verify and validate the created models, a hot rod rolling problem, a blow pipe problem, a dam network design problem, a crime rate prediction problem and a supply chain design problem are modeled and implemented.

There are several potential contributions as a result of completion of this dissertation as follows:

With respect to new knowledge

- Mathematics for multi-level SM selection and ensemble of SM
- Mathematics for partitioning-approximation-coordination of complex design problems

With respect to functionality and Utility

- Enabled SM selection easier and faster than existent selection methods by creating a systematic and organized automated selection process
- Enabled prediction with sparse data by ensemble of SM
- Enabled dealing with temporal and spatial data in SM
- Enabled complexity management in a supply chain network by partitioning the network, approximating the solutions of the partitions, and coordinating them

9.2 Contributions

In Figure 9.1, the dissertation outline and the work accomplished during this PhD is summarized.

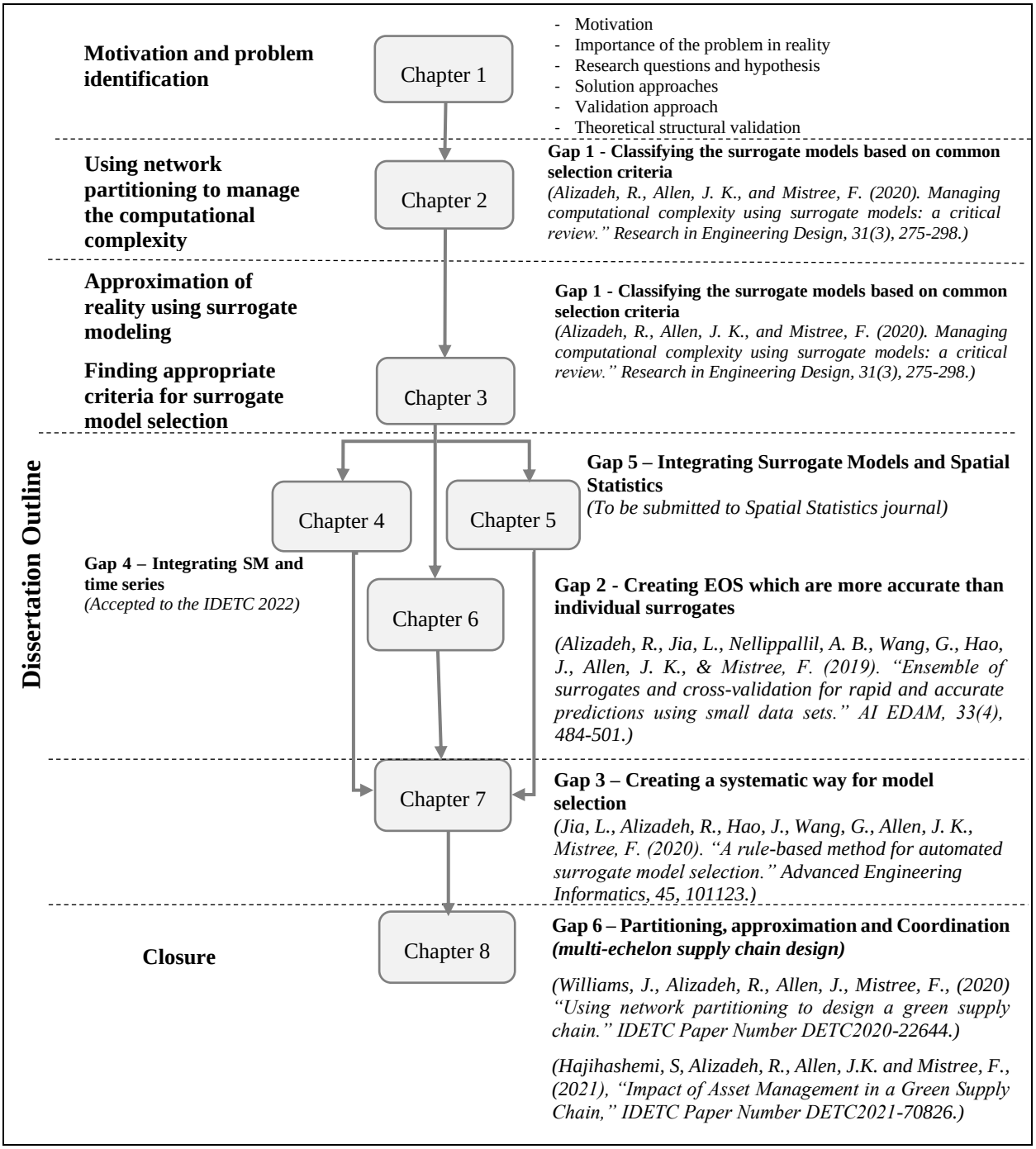


Figure 9.1. The dissertation outline and the work accomplished during this PhD.

As shown in Figure 9.2, Work 1 is accomplished through creating a framework to provide guidance for researchers and practitioners to choose the most appropriate surrogate model based on incomplete information about an engineering design problem. A visual description of this framework is shown in Figure 9.2.

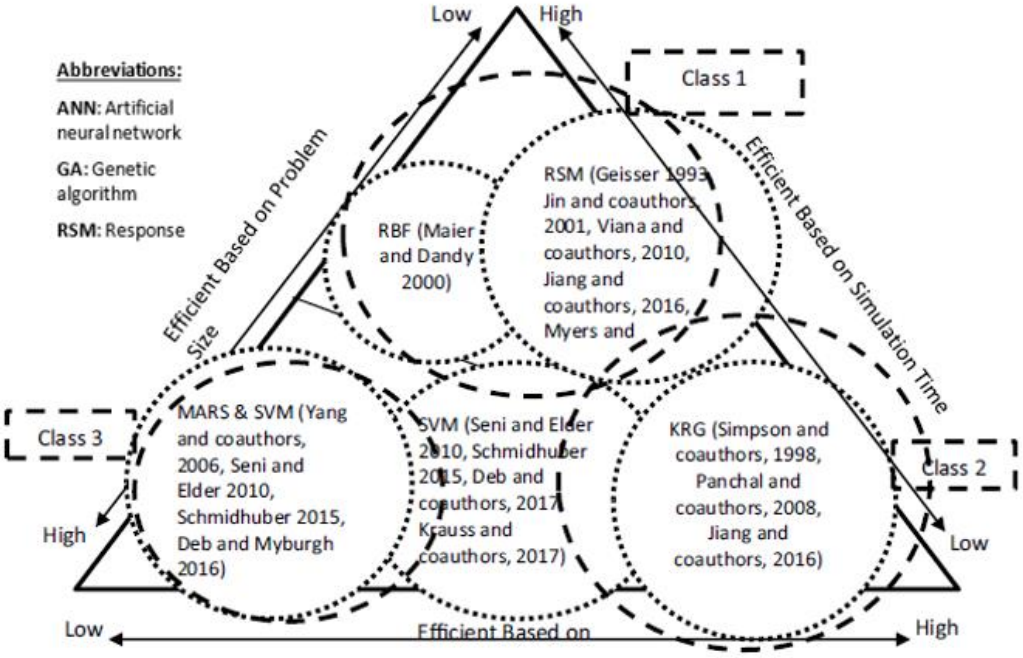


Figure 9.2. A visual description of the framework to provide guidance for researchers and practitioners to choose the most appropriate surrogate model.

The framework shown in Figure 9.2, is a graphical representation of the key outcome of addressing Gap 1, which is on classifying the surrogate models based on common selection criteria. To achieve this outcome and address the gap, the following research question is posed "What are the main classes of the design of experiment (DOE) methods, surrogate modelling methods and model-fitting methods?". The hypothesis is that Surrogate modelling methods can be classified based on the problem size, computational time, and accuracy. This hypothesis is tested and proved to be correct, and the outcome is published in (Alizadeh et al., 2020a).

Work 2 in Figure 9.3, which is the content of Chapter 6, is accomplished through creating an approach to build an ensemble of surrogates (EOS) which is more accurate than individual surrogates. In Chapter 6 the key outcome of Work 2 is articulated in answering the question of "What is the mathematics to create EOS that are more accurate than individual surrogate models?". Through this work it is hypothesized that using a weighted average of individual surrogate models and minimum overall cross validation error such EOS can be built. The hypothesis is proved to be correct through testing the proposed approach on a hot rod rolling problem as the EOS is relatively computationally inexpensive and faster than the individual surrogates. In Chapter 7 of this dissertation, the research question of "What is the mathematics to increase the interpretability and manage the computational expense?" is answered through testing the "using an interpretable decision tree to map problem scale, noise, size of sample and nonlinearity to the types of SM and select the promising SM and function type; then, using a genetic algorithm to find the appropriate hyper-parameters for each selected SM". This is Work 3 in Figure 9.3, which is resulted in drastic increase in the selection pace by pre-screening of surrogate model types based on selection rule extraction and published in (Jia et al., 2020). A graphical representation of the approach proposed in Work 3 is shown in Figure 9.3.

In Chapter 4, Gap 4 is addressed which is on integrating the surrogate models and time series analysis. To address this gap, the research question of "What is the appropriate SM to use when the data includes time-dependent variables as predictors?" is answered. It is hypothesis that through replacing the design of experiments with the time lags analysis, we find the SM which is useable in temporal variables. These SMs are better than classic time series analysis methods like ARIMA, MARIMA and ARIMAX. Also, EOS are better than individual SM in temporal variables."

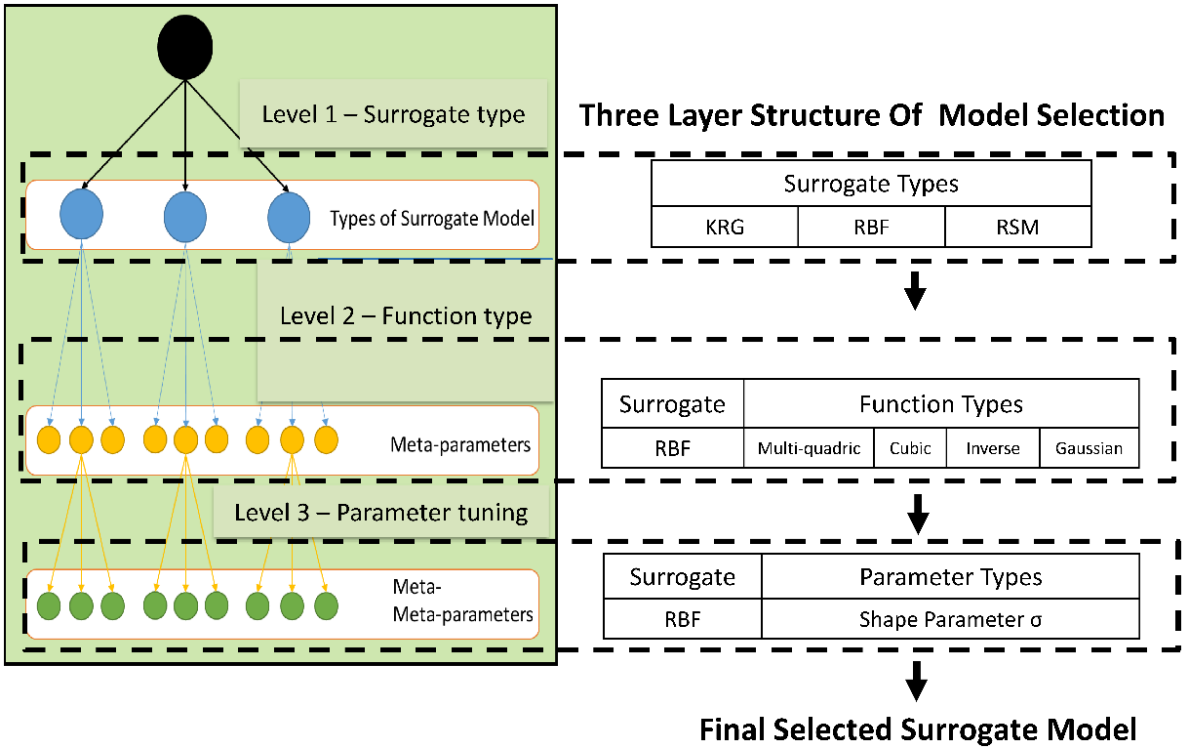


Figure 9.3. A visual representation of the proposed approach for AutoSM selection in Chapter 7.

The hypothesis is tested through employing temporal-based SM on an example of water network design and dam inflow water prediction. Based on the results, time series integrated SM are relatively less computational expensive and more rapid compared to classic time series analysis models. The key outcome of addressing Gaps 4, is enabling to dealing with temporal data by incorporating time series (lag analysis) with SM. Temporal data can be used in SM using different time lags.

In Chapter 5, addressing Gap 5 on integrating surrogate models and spatial statistics is discussed. To address this gap, a research question of "what is the appropriate SM to use when the data includes spatial variables as predictors?" is posed and it is hypothesized that by replacing the design of experiments with the geographically weighted correlation analysis, the surrogate model

which is useable with spatial variables can be found. The key outcome of addressing this gap is enabling the use of spatial statistics and particularly the concept of the geographically weighted regression in surrogate modeling, to create spatial surrogate models.

Gap 6, which is on partitioning, approximation, and coordination of complex engineering design problems to manage the computational complexity. This gap is dealt with through Works 6 and 7 and answering the research question of "What is the mathematics to partition, approximate and coordinate a complex multi-channel and multi-echelon supply chain design problem?".

The hypothesis is that using a bi-level partitioning approach, the two different channels (online and in-store shopping) can be modeled, then a surrogate model can be used to map the solutions of the lower-level into the upper-level. A partitioning-approximation-coordination framework is proposed to deal with the computational complexity and uncertainty of engineering design problems.

To verify and validate the proposed partitioning-approximation-coordination framework, a multi-echelon, multi-channel green supply chain problem is designed and tested. Since this problem is a comprehensive problem to verify the both approximation (surrogate modeling portion of this dissertation) and partitioning-coordination (second portion of this dissertation), it is divided into two parts: (i) two-level partitioning to design a green supply chain (two-echelon supply chain design), where a mixed-integer programming is used to design the supply network and a k-means method used to partition and coordinate the network. (ii) partitioning, approximation, and coordination to manage complexity to design the multi-echelon portion of the problem as well as multi-channel portion of the supply chain design, where two channels including online and in-store shopping are modelled.

The key outcomes of accomplishing these two works are creating a framework to partition-approximate-coordinate complex design problems and a method for calculating the difference in expected carbon emissions between in-store shopping versus e-commerce-based online retailing involving last mile delivery to customers' homes to quantify which channel has the least harmful impact on the environment.

9.3 I Statement

In this chapter, I plan to determine if the objectives planned for the dissertation are addressed. I plan to carry out a self-reflection of what have been achieved in past chapters and identify enabling technologies that requires advancement to further develop the vision of managing computational complexity and uncertainty in engineering design.

In this chapter, the I summarize some of the key concepts that form the basis of evolving cyber-physical social systems transitioning from push to pull low carbon economy. Finally, my vision for research in systems-based design architecture is addressed from the context of an evolving cyber-physical social systems transitioning from push to pull low carbon economy. I have taken away important lessons as following:

- Identify domains that are foundational to advance the vision of managing computational complexity and uncertainty in engineering design
- Opportunities for improving the current tools, frameworks, and algorithms for multi-echelon, multi-channel, green supply chain design under uncertainty
- Possible future directions for systems-based design that intersects with the emerging trends in the field of evolving cyber-physical-social systems

- Identification of gaps and future vision in evolving cyber-physical-social systems transitioning from a push economy to a low carbon pull economy
- Developments needed in academic curriculum to address evolving cyber-physical-social systems in an online, internet-based pull economy in a socially responsible, low carbon economy
- Advances needed in industrial scale to address requirements to cope with evolving cyber-physical-social systems in an online, internet-based pull economy in a socially responsible, low carbon economy

During my chapters 1 to 8, I realized that supply networks and other network-based systems are more vulnerable to uncertainties due to their decentralized natures. Therefore, uncertainty management is critical to preserving acceptable performance. Besides, network-based systems are complex, and modeling them is computationally expensive.

Thus, complexity management is essential to model and develop innovative solutions to improve the performance of these networks. I worked on both uncertainty management and complexity management in complex network design problems to provide the answers to the following research question:

- "What is the mathematics to build prediction models that are more accurate, rapid, less computationally expensive under conditions of multi-dimensional sparse data?" where I practiced data analytics in production systems and created a mathematical approach to build an ensemble of predictive models (called surrogate models) to deal with sparse data in a rapid and accurate way. I also used time series analysis and spatial statistics in my predictions.

- “What is the mathematics to automatically select the best predictive model for a given prediction problem?” where I created a rule-based, multilevel, automatic surrogate model selection approach for automated machine learning (AutoML). The drastic increase in the selection pace by pre-screening of surrogate model types based on selection rule extraction is the scientific contribution of the proposed method.
- “What is the mathematics to partition a complex network problem, then approximate the problem, and coordinate the solutions of partitions to obtain the solution of the whole problem?”

Summing up, my dissertation falls in the realms of data science and analytics and operations research and is on managing complexity and uncertainty in complex network design considering environmental sustainability; in applications including but not limited to CPS systems including healthcare supply chains, energy systems, healthcare systems design and analytics, transportation networks, complex systems science and modeling and smart manufacturing, in a low carbon economy. Based on my Ph.D. research, my vision is to design, evaluate, predict, and manage evolving network-based, resilient, and sustainable CPS systems. To fulfil this vision, I plan to focus on automating the modeling, model evaluation, and model selection/combination processes by using data-driven approximation algorithms for capturing different dimensions of an evolving CPS system.

9.4 Research Thrusts and Applications

Building on my dissertation, in my early career (2022-2026), I plan to seek answers to the key challenges anchored in two Research Thrusts: RT1 and RT2. Each RT is an extension of my doctoral research and comprises several research foci as shown in Figure 9.4.

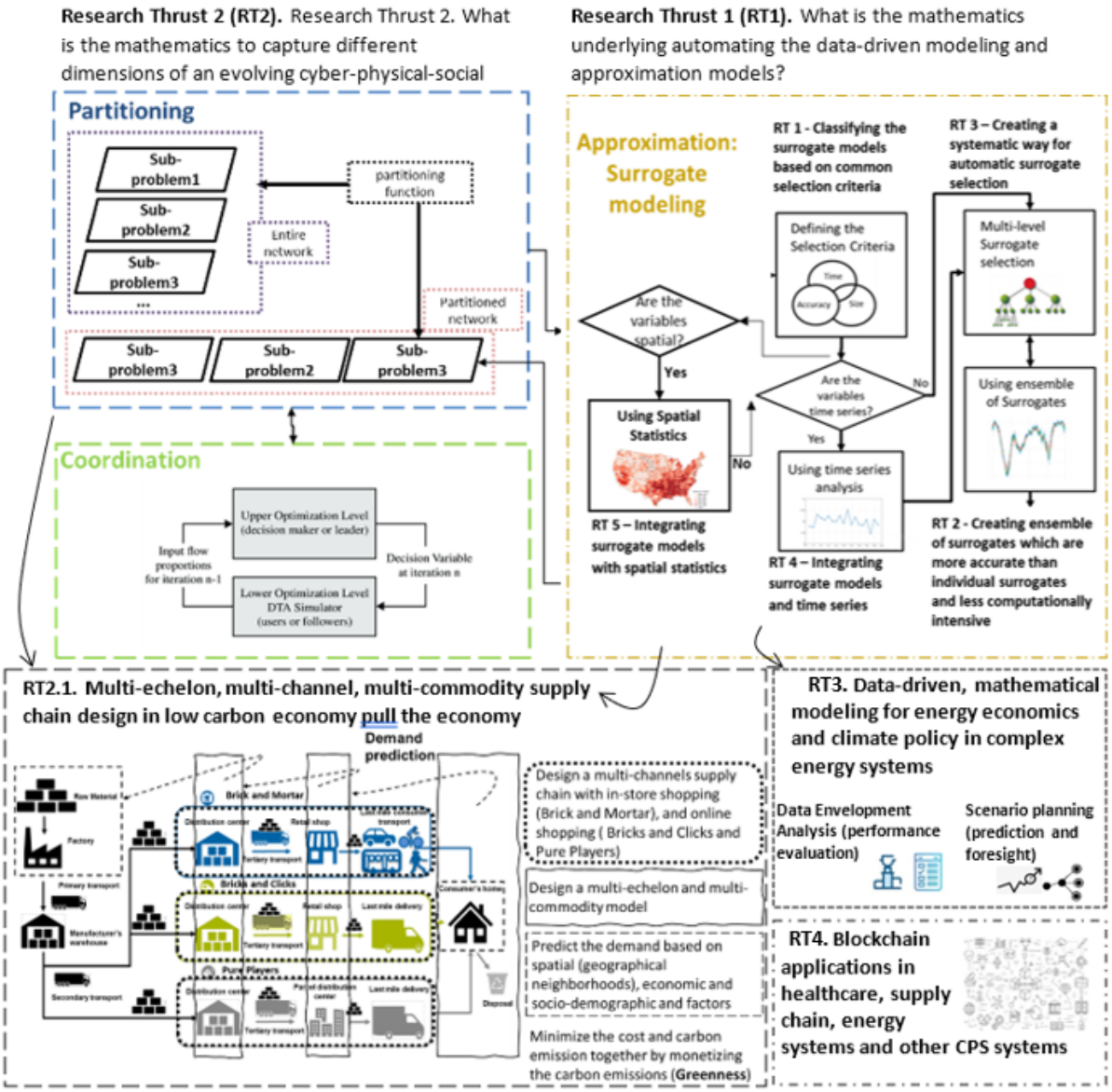


Figure 9.4. Research thrusts.

9.4.1. RT 1. What is the mathematics underlying automating the data-driven modeling and approximation models?

This research thrust has four main foci as follows:

1. The first focus is on creating rule-based computational frameworks, knowledge-based platforms, and ontologies to manage the knowledge resulting from the modeling, simulation, and approximation process and *automate* the selection of the best data-driven model based

on modeling time, desired accuracy, and data dimension criteria. I laid out the foundation for this focus in (Alizadeh et al., 2020a).

2. The second focus is on improving and integrating decision models leveraging new technologies such as the Internet of Things (IoT), cloud computing, and deep learning to support managing system complexities and dealing with the evolution of the systems. The evolution of the systems happens due to the emergent properties, which are one of the origins of uncertainty when using the most appropriate data-driven models manage the evolution of such uncertainty.
3. The third focus is on building statistical models to replace computationally expensive simulation models using different features, including time, geographical locations, textual, visual, and other unlabeled features. I laid out the foundation of this focus in (Alizadeh et al., 2020a; Jia et al., 2020).
4. The fourth focus is on creating mathematics to build a combination of models in two different situations where:
   (i) We are dealing with sparse data; here, I am looking at adaptive sampling methods, including methods based on cross-validation errors. This focus aims to build more accurate and less computationally expensive models. I laid out the foundation for this focus in (Alizadeh et al., 2019).
   (ii) The data is abundant, and we deal with big-data analytics; here, I am looking at building models for big data analytics and large-scale data mining such as deep learning. I laid the foundations of this focus in (Alizadeh et al., 2020a; Jia et al., 2020).

The fifth focus is on curating synthetic data when the data is limited and sparse and generating and collecting the data are non-trivial. Generating and collecting data in engineering design is

computationally expensive. Also demand for access to data, especially data collected using public funds, is ever growing. At the same time, concerns about the disclosure of the identities of and sensitive information about the respondents providing the data are making the data collectors limit the access to data. Furthermore, building and testing machine learning models requires access to large and diverse data. However, finding usable datasets without running into privacy issues is cumbersome. This objective in this focus is to create techniques for generating synthetic data—fake data generated from real data—so we can perform secondary analysis to do research, understand customer behaviors, develop new products, or generate new revenue. Synthetic data sets, generated to emulate certain key information found in the actual data and provide the ability to draw valid statistical inferences, are an attractive framework to afford widespread access to data for analysis while mitigating privacy and confidentiality concerns. This helps data scientists to learn how synthetic data generation provides a way to make such data broadly available for secondary purposes while addressing many privacy concerns. This also helps analysts to learn the principles and steps for generating synthetic data from real datasets. Also, business leaders see how synthetic data can help accelerate time to a product or solution.

In this research thrust, I follow:

- the steps for generating synthetic data using multivariate normal distributions;
- methods for distribution fitting covering different goodness-of-fit metrics;
- how to replicate the simple structure of original data;
- an approach for modeling data structure to consider complex relationships;
- multiple approaches and metrics you can use to assess data utility;
- how analysis performed on real data can be replicated with synthetic data;
- privacy implications of synthetic data and methods to assess identity disclosure

9.4.2. RT 2. What is the mathematics to capture the dimensions of an evolving CPS system and the dynamics among them?

The objective in Research Thrust 2 (RT2) is to manage complexity and uncertainty using the approximation-partitioning-coordination framework. For example, a green supply network can be partitioned into partitions that are supplied by a distribution center and coordinated using a mixed-integer, bi-level optimization model, where the first level is to design the best topology for the chains (partitioning the region of supply into smaller regions to be supplied by one distribution center) and the second level is to find the best route for the delivery vans in online shopping. Then, to manage the computational complexity of the bi-level model, the lower level can be approximated through a surrogate model. Also, in many complex CPS system studies, geographical correlations of the components are overlooked. However, spatial correlations have an implicit but critical impact on the system components' behavior and the systems' evolution. The foundations of this focus have been laid out in (Hajihashemi et al., 2021; Williams et al., 2020).

9.4.3. RT 3. Data-driven, mathematical modeling for energy economics and climate policy in complex energy systems

The third research thrust is on data-driven mathematical modeling for complex CPS networks, including but not limited to energy networks, healthcare networks, supply networks, and production networks. I plan to use Input-Output models and Data Envelopment Analysis (DEA) to evaluate the performance of CPS networks. I laid the foundations to conduct this sort of research in (Alizadeh et al., 2020b; Gharizadeh Beiragh et al., 2020; Kaleibari et al., 2016). I also plan to use multi-criteria decision-making models to identify, classify and rank the interdependencies among decision-making criteria to prioritize and allocate the resources to the most critical operation in supply network operations, healthcare logistics, and supply chain, and energy networks. The foundations of this focus have been laid out in (Alizadeh et al., 2020d). Besides, I

aim to use data-driven models to simulate and predict the greenhouse gas emission and environmental recourses consumption in different projects and provide policymakers with recommendations and managerial insights to take critical strategies to achieve sustainable development goals. In (Gharizadeh Beiragh et al., 2020; Soltanisehat et al., 2020; Zamani Sabzi et al., 2018), the foundations of these predictive models in the form of time series analysis and mixed-integer optimization models have been laid out. Furthermore, besides using short-term prediction to manage typical uncertainties, I use scenario-based planning to deal with disruptions and deep uncertainties. I have created models for this type of foresight in (Alizadeh et al., 2016b; Alizadeh et al., 2020c).

9.4.4. RT 4. Blockchain applications in healthcare, energy, supply chain and other CPS systems

Blockchain can be used to build a peer-to-peer, secure, and smart transaction system. As a horizontal technology that has changed several fields of industry, blockchain has tremendous potential to transform healthcare systems as well.

Besides the benefits of blockchain for healthcare systems, blockchain has shortcomings in terms of security and scalability, which opens future research directions. Combining blockchain with artificial intelligence and cloud computing is one of the most interesting and probably demanding directions for solving issues with blockchain decentralization which I am planning to explore. Besides, there are opportunities for a number of use cases ranging from emerging peer-to-peer (P2P) energy trading and Internet of Things (IoT) applications, to decentralized marketplaces, electric vehicle charging and e-mobility in a low carbon pull economy that I explore.

9.4.5. RT 5. Social aspect of the cyber-physical-social systems

In this dissertation, the foundation of investigating a cyber-physical system was laid out. However, the social dimension which is very important and influential in designing the cyber and physical aspects was not studied. Therefore, the social aspect will be studied as a future research thrust. In Figure 9.5, the elements of Socio-Cyber-Physical (SCP) Systems are introduced, where human stakeholders play a prominent role. Here, technological aspects of the systems are developed such that they support the involved human stakeholders in a sustainable way.

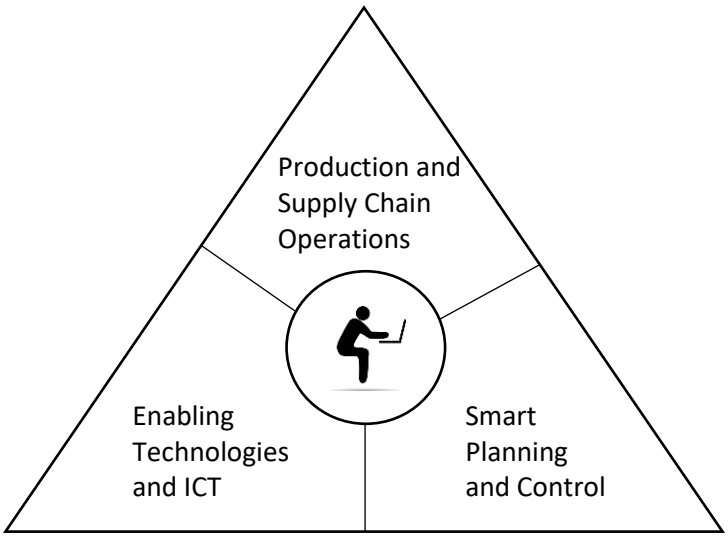


Figure 9.5. Components of a socio-cyber-physical system.

Global production networks and hence linked Socio-Cyber-Physical Systems involve the interaction of individual decision-makers from different contexts, acting in interdependent organizations. The competitiveness of the resulting production network depends on the capability to bridge technical differences as well as the introduced context-dependent behavioral differences between human stakeholders. Here, managing existing social idiosyncrasies of linked SCP is critical for sustainable, accountable, and competitive production networks. Figure 9.6 shows a network of linked SCP, where each system involves local human stakeholders.

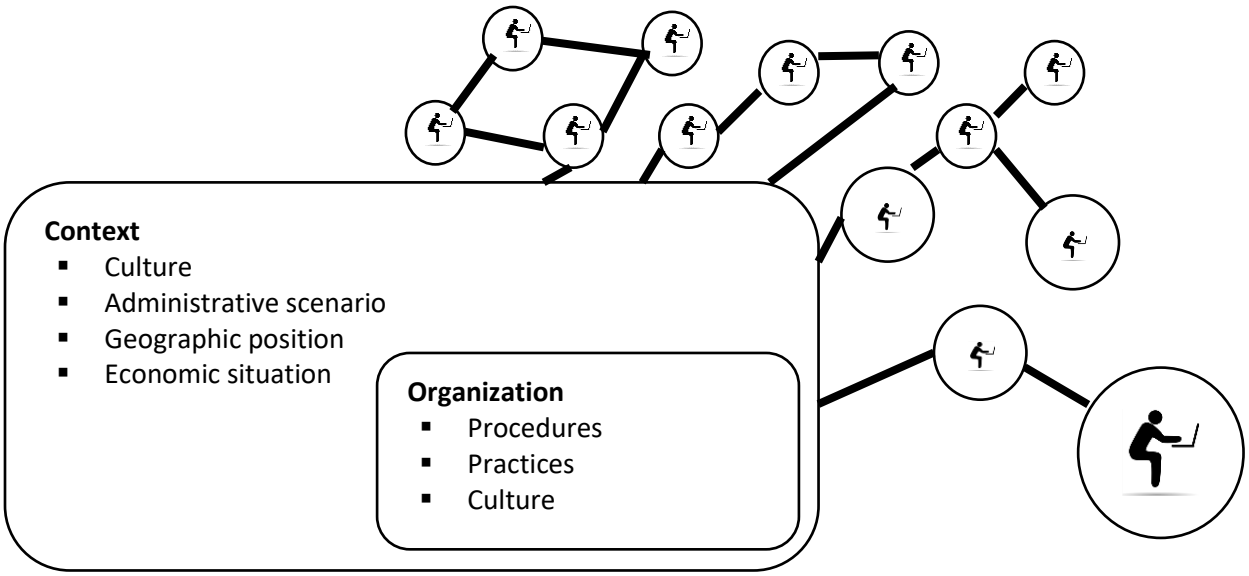


Figure 9.6. Linked socio-cyber-physical systems forming a production network.

The transformation of production networks into networks of Socio-Cyber-Physical Systems requires a comprehensive design approach. This approach comprises findings of the research stream dedicated to technical aspects of CPS as well as a new stream of research for the implications of context-dependent behavioral aspects of human stakeholders.

This stream needs to develop models, measures and tools for aspects related to the embedding of human stakeholders with different individual, organizational and contextual backgrounds (Frazzon, 2009; Scholz-Reiter et al., 2010a; Scholz-Reiter et al., 2010b).

9.4.6. RT 6. Environmental sustainability in the cyber-physical-social systems

The last decades have also seen a marked increase in the interest of the political and academic world and public opinion regarding the sustainability of human activity. In this context, the sustainable impact of business activity can be positively influenced by the green energy usage, green production and supply chain if included in a life-cycle oriented strategy. As shown in Figure 9.7, embedding the environmental sustainability can be through the supply chain and logistics

process (transportation and material handling), transportation (different modes of transportation in logistics and supply chains, including renewable and sustainable energies and fuels – for example, electric vehicles, biodiesel, drones' delivery, air taxies, etc.), manufacturing (using automation, clean energies in the process and carbon capture and storage technologies), production systems, material handling within the inventories, warehouses, distribution centres and stores.

Besides, energy production systems, transmission, and distribution process are major areas of GHG emission and managing these stages is an ongoing, unsolved, and broad area of research. I laid out the foundation of this research thrust in (Alizadeh et al., 2020b; Alizadeh et al., 2016a; Alizadeh et al., 2016b; Alizadeh et al., 2020c; Alizadeh et al., 2016c; Alizadeh et al., 2015a; Alizadeh et al., 2014; Alizadeh et al., 2015b; Alizadeh and Soltanisehat, 2020; Alizadeh et al., 2020d; Beiragh et al., 2020; Gharizadeh Beiragh et al., 2020; Kaleibari et al., 2016; Shafiei Kaleibari et al., 2016; Soltanisehat et al., 2020; Soltanisehat et al., 2018; Zamani Sabzi et al., 2018).

9.4.1. RT 7. Green supply chain as a cyber-physical-social system using digital threads

In this research thrust, I answer the following questions: (i) how can seamless visibility across the supply chain enable informed decision making? (ii) how can prescriptive decisions enable supply chain planners to navigate through unforeseen and exceptional scenarios?

Besides, as shown in

Figure 9.8, I build digital threads to deal with the unexpected events that can affect the supply chain.

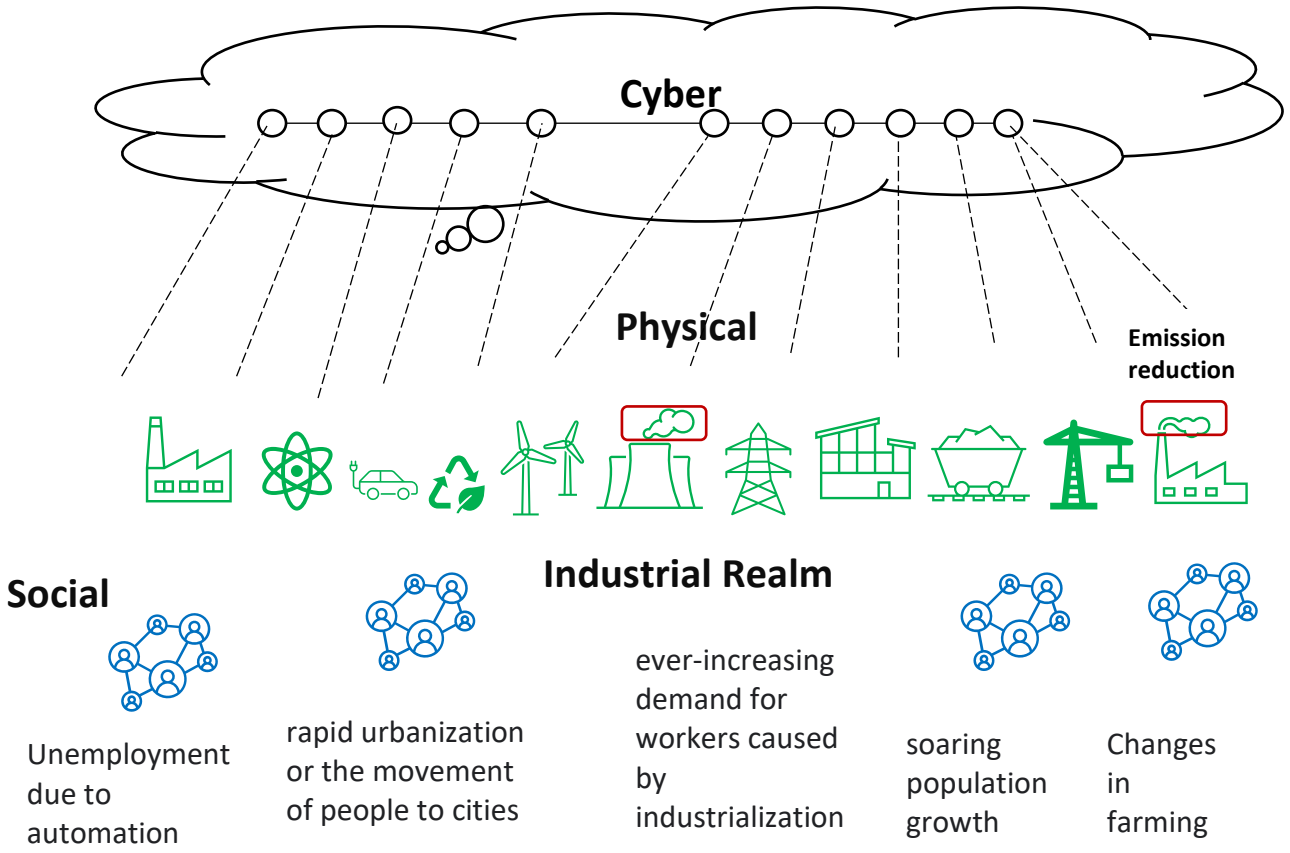


Figure 9.7. Some social aspects of cyber-physical-social system.



Figure 9.8. Unexpected events affecting the supply chain.

In Figure 9.9, the impact of unexpected events on supply chain (SC) is shown in detail. Also, it is shown that the digital thread is a way to oversee and supervise the supply chain by sensing the system through sensors, monitoring the changes, and collecting data over time, analyze and simulate the possible scenarios, and acting before a disruption occurs.

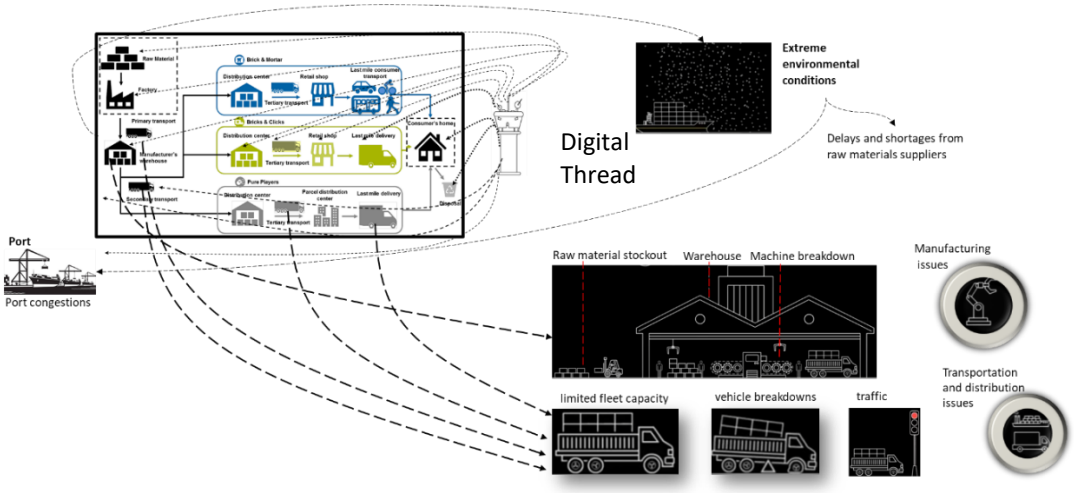


Figure 9.9. Impact of unexpected events on SC and how digital thread monitors them.

This process is shown in Figure 9.10.

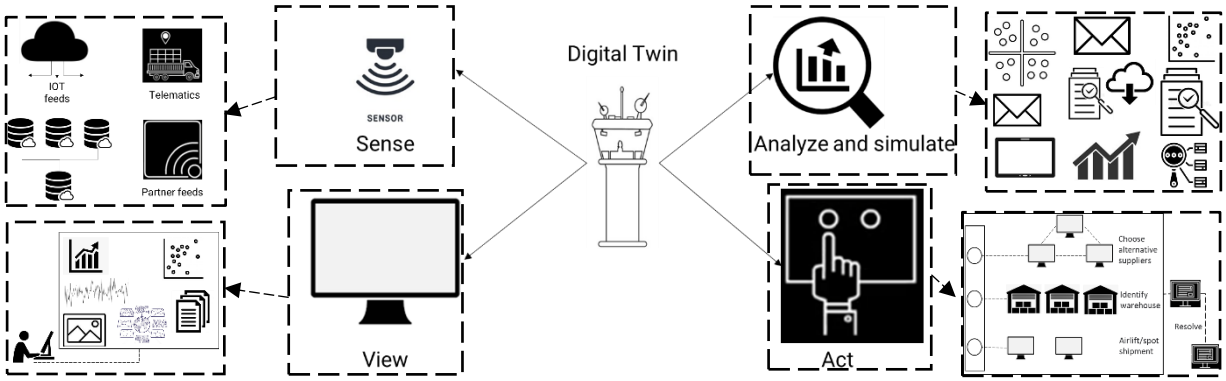


Figure 9.10. How digital thread works.

In the sense step, using digital thread the demand can be sensed and a proactive service risk alert can be set via simulated supply chain. In view step, labelled data, images, temporal, and spatial data as well as alerts and indicators are monitored and what if scenarios are investigated.

In step Analyze and simulate, different scenarios are conducted and analyzed to reallocate inventory across product lines, recommend target and safety stock, plan the production process, select the alternative suppliers, optimize the real time route, investigate the alternative fleet options and other what if scenarios. Finally, in the fourth step, trigger order creation for high demand product, component quality alert orchestration, and production planning execution are performed. Through using digital threads in supply chains, several goals and performance enhancement can be achieved as shown in Figure 9.11.

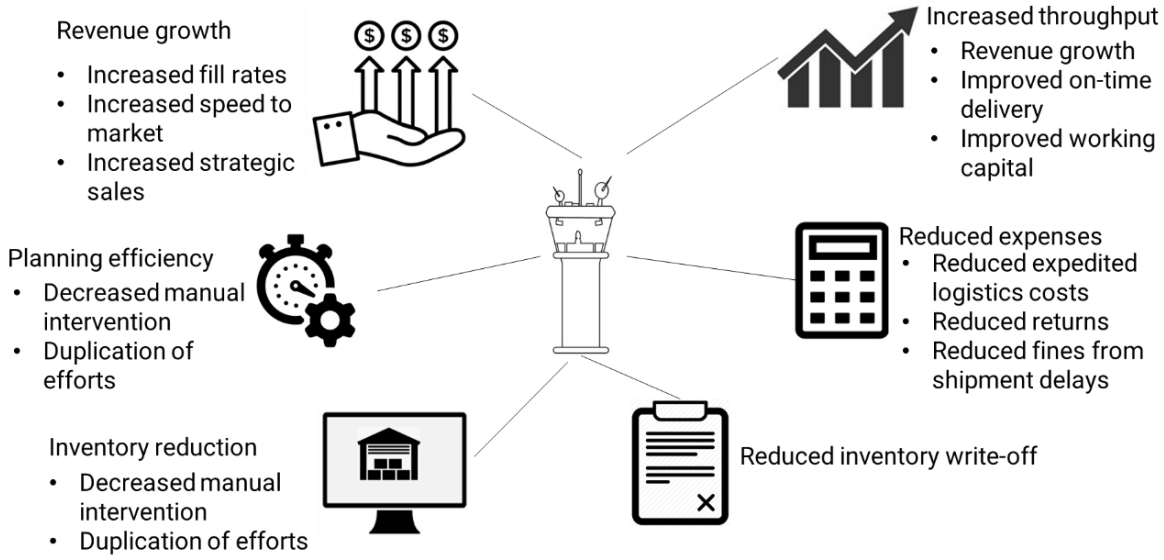


Figure 9.11. The benefits of using digital threads in supply chains.

As shown in Figure 9.12, there are two main phases in the evolution of the digital threads in supply chains, including linear supply chain and network ecosystem while there are four analytical approaches, including descriptive, predictive, prescriptive, and autonomous. In the descriptive approach, functional E2E ecosystem visibility and synchronous collaboration, clear reporting and analytics, and balance cost optimization, risk, mitigation, and growth are the foci. In the predictive approach, the foci are integrated E2E visibility with cross-functional real-time dashboards, synchronized parameter setting and optimization, early warning system with clearly defined alerts,

and respond to global shocks and shifting customer demands. In the prescriptive approach, the aim is to design flexible, cost effective and resilient supply chain ecosystems, prescriptive recommendations based on AI, scenario capabilities spanning the entire supply chain, and integrated optimization tools to optimize key supply chain parameters in real time.

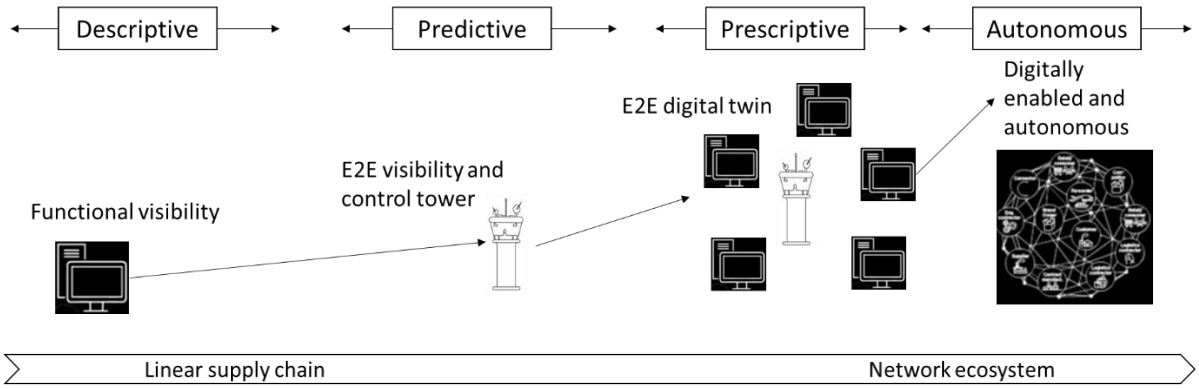


Figure 9.12. The evolution of digital threads in supply chains.

Finally, in the autonomous approach, the objective is to self healing master data and planning parameters, digital twin orchestration of supply chain with continuous planning, and enterprise level digital twin.

9.5 Overall Theme of my Research based on my PhD

The overall theme of my research is projecetd in Figure 9.13. Based on the foundations that I have laid out in my dissertation and the papers i have published, i am planning to work on my NSF research proposal on designing, evaluating, predicting, and managing evolving cyber- physical-social systems in low-carbon pull-push economy.

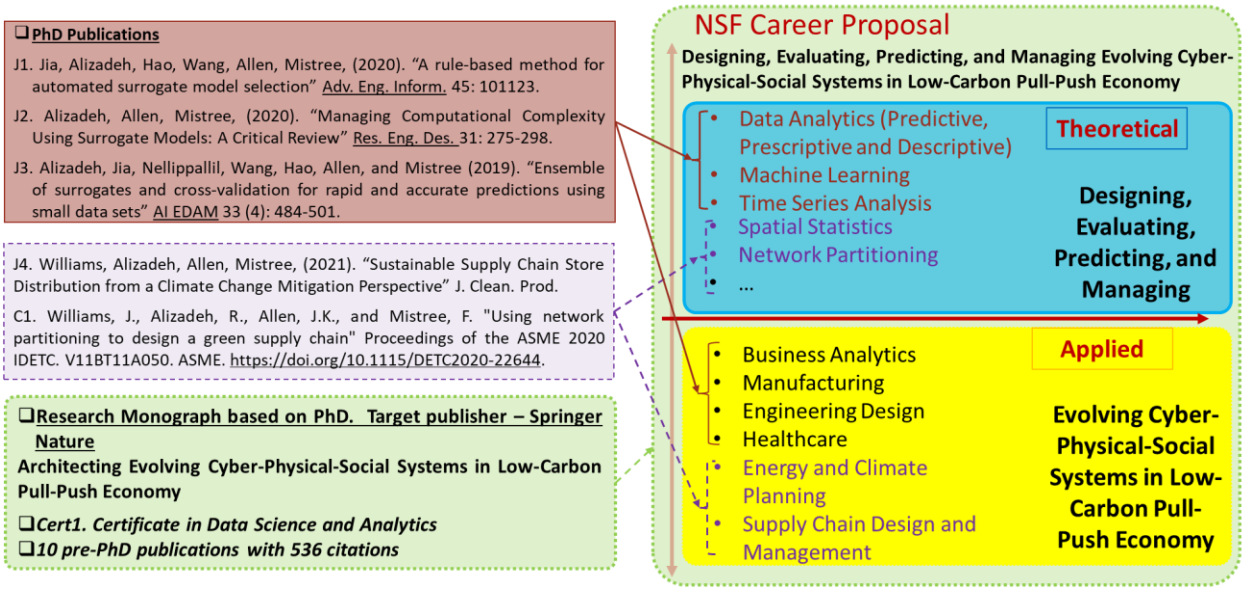


Figure 9.13. Future work based on my PhD.

Appendices

Appendix A. Surrogate Models (Section 7.4.1)

1. Response Surface Method

Response surface method (RSM), which is also known as polynomial regression method (PRM), has the simplest type of parameters (that is coefficients in a polynomial function). It is objectively calculated through the least square regression method (Razavi et al., 2012). In most of RSM problems, the limited pre-knowledge about the relationship between response variables and factors severely control the widespread use of model and its performance. Hence, how to select the order of RSM becomes worthy to investigate challenge. So, in this dissertation, the order of RSM is considered as the optional parameter, named '***Order***'. If we presume $\boldsymbol{x}$ as an independent vector of factors, $\boldsymbol{y}$ is the vector of response, the notation follows through subsequent surrogates. The impact of $\boldsymbol{x}$ on $\boldsymbol{y}$ and their relationship can be illustrated as follows:

$$\hat{\boldsymbol{y}}(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{V}(\boldsymbol{x}|Order) \qquad (A.1)$$

where $\boldsymbol{w}$ is the coefficient vector of the polynomial function. '***Order***' represents the max number of interacting factors, $\boldsymbol{V}(\boldsymbol{x}|Order)$ denotes a vector of interacting factors corresponding to different orders. For example, when order=1, $\boldsymbol{V}(\boldsymbol{x}|Order = 1) = [1, x_1, x_2 \dots, x_d]^{\boldsymbol{T}}$**.** d is the number of input factors; when order =2,

$$\boldsymbol{V}(\boldsymbol{x}|Order = 2) = [1, x_1, x_2 \dots, x_d, x_1x_1, x_1x_2, \dots, x_1x_d, x_2x_2 \dots, x_2x_n \dots, x_nx_n]^{\boldsymbol{T}} \qquad (A.2)$$

In this section, we select '***Order=2,3,4,5,6***' RSM as part of our candidate surrogate models.

2. Kriging

Kriging is an interpolating technique, which consists of a polynomial model of an input vector of factors $\boldsymbol{x}$, $f(\boldsymbol{x})$, and localized deviation of $\boldsymbol{x}$**,** $R(\boldsymbol{x})$, as follows:

$$\hat{y}(\boldsymbol{x}) = f(\boldsymbol{x}) + R(\boldsymbol{x}) \qquad \text{(A.3)}$$

$f(\boldsymbol{x})$ is the polynomial term, which is a global function over the entire input space (Razavi et al., 2012). Normally, $f(\boldsymbol{x})$ is set as a constant number or a linear polynomial function and $R(\boldsymbol{x})$ is a localized deviation function. In this article, our focus is on one type of Kriging, ***Ordinary Kriging***, in which it is assumed that,

$$\hat{y}(\boldsymbol{x}) = \mu + R(\boldsymbol{x}) \qquad \text{(A.4)}$$

where μ is an unknown constant value, which represents the simulation output averaged over the 'experimental area. $R(\boldsymbol{x})$ is a zero-mean stochastic process with known covariance function,

$$\mathrm{C}_{ij} = \mathrm{C}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{Cov}\left[R(\mathbf{x}_i), R(\mathbf{x}_j) \right] = \sigma^2 R_{ij} \qquad \textit{(A.5)}$$

R_{ij} is the correlation function with the i^{th} and j^{th} data point. σ^2 denotes the variance of this stochastic process. In addition, the hyper-parameter, θ of each correlation function is determined by solving the nonlinear hyper-parameter optimization problem.

3. Radial Basis Function

This technique is a mathematical function and which accepts real values and its value calculated based on the distance the origin and each point (Montgomery, 2017). Also, the distance between the center point and each point can be used as an alternative as shown in eq. (A.6).

$$r_{i,j} = r(\boldsymbol{x_i}, \boldsymbol{x_j}) = \left\| \boldsymbol{x_i} - \boldsymbol{x_j} \right\| \qquad \textit{(A.6)}$$

Where $r_{i,j}$ denotes the Euclidean distance between two different point in the samples. Besides, some radial functions are employed to connect the distance $\boldsymbol{w} = [w_1, w_2 \ldots w_M]$ with the outputs, then the integration of these functions is used to estimating the complicated mathematical functions. We can use these functions in constructing the surrogate models like eq. (A.7).

$$\hat{y}(x_{new}) = \sum_{i=1}^{M} r_i Q(\|x_{new} - x_i\|) \qquad (A.7)$$

where the surrogate function $\hat{y}(\boldsymbol{x_{new}})$ stands for an integration of radial basis functions and each of them is linked to a basis function $Q(\|\boldsymbol{x_{new}} - \boldsymbol{x_i}\|)$ and has a weight of w_i (Broomhead and Lowe, 1988).

4. Multivariate Adaptive Regression Splines

This section presents a brief introduction of the MARS surrogate model, the detailed technique is described by (Friedman, 1991). MARS is a nonparametric approach based on partitioning data sets into separate segments, and adaptively selecting a set of basic functions to approximate these segments through a forward/backward recursive approach. A MARS model is supposed to describe with eq. (A.8).

$$\hat{y}(\boldsymbol{x})=a_0 + \sum_{m=1}^{M} a_m B_m(\boldsymbol{x}) \qquad (A.8)$$

where $\boldsymbol{a} = a_0, a_1, \dots, a_M$ denote the constant coefficients, estimated using the least-squared method. Each $B_m(\boldsymbol{x})$ is a BF, which can be represented by eq. (A.9).

$$B_m(\boldsymbol{x}) = \prod_{k=1}^{K_m} \left[s_{k,m} \left(x_{v(k,m)} - t_{k,m} \right) \right]_+^q \qquad (A.9)$$

Here K_m is the number of factors in the m^{th} basis function, $s_{k,m} = \pm 1$, $x_{v(k,m)}$ denotes the v^{th} variable, $t_{k,m}$ is the knot value corresponding to the v^{th} variable, which is regarded as the interface points between each subsample piece. The subscript '+' and superscript 'q' represent the function as a truncated power function, of power is 'q'.

Piecewise linear function follows an exact form $max(0, x - t)\ or\ \max(0, t - x)$, so the spline function $\left[s_{k,m} \left(x_{v(k,m)} - t_{k,m} \right) \right]_+^q$ can be expressed as

$$\left[s_{k,m}\left(x_{v(k,m)}-t_{k,m}\right)\right]_+=\begin{cases}max\left(0,x_{v(k,m)}-t_{k,m}\right)=\begin{cases}x-t,\ \ if\ x\geq t\\ 0,\ otherwise\end{cases}\\ \boldsymbol{or}\\ max\left(0,t_{k,m}-x_{v(k,m)}\right)=\begin{cases}t-x,\ \ if\ t\geq x\\ 0,\ otherwise\end{cases}\end{cases} \qquad (A.10)$$

Also, the basic functions, formally, can be a single spline function, or the product of two or more spline functions associated with different variables. The number of interacting spline functions is determined artificially and is called '***Max_Interaction***'.

Appendix B. Theoretical problems (Section 7.4.2)

We introduced the selected benchmark functions to use in this chapter in Chapter 7.

1. Function of Perm

$$f(\boldsymbol{x})=\sum_{k=1}^{n}\left\{\sum_{j=1}^{k}(j^k+0.5)^2\left[\left(\frac{x_j}{j}\right)^k-1\right]\right\}^2 \qquad (B.1)$$

$where\ x_j\epsilon[-n,n], i=1,\dots,n, \qquad n=10$

2. Dixon and Price function:

$$f(\boldsymbol{x})=(x_1-1)^2+\sum_{i=2}^{n}i(x_i{}^2-x_{i-1})^2 \qquad (B.2)$$

$where\ x_i\epsilon[-10,10], i=1,\dots,n, \qquad n=5$

3. Beale function

$$f(\boldsymbol{x})=(1.5-x_1+x_1x_2)^2+(2.25-x_1+x_1x_2{}^2)^2+(2.625-x_1+x_1x_2{}^3)^2 \qquad (B.3)$$

$where\ x_1,x_2\epsilon[-4.5,4.5]$

Appendix C. Benchmark Functions and DoE (Section 7.5)

We introduced the selected benchmark functions to use in this dissertation in Chapter 7.

1. Benchmark Functions

We present results based on 10 instances of the design of experiments for all the problems. 1-4 benchmark functions are referred from (Mehmani et al., 2018), 5-9 benchmark functions are referred from (Jin et al., 2001).

2. Brain-Hoo function (BH)

$$y(\boldsymbol{x}) = \left(x_2 - {5.1{x_1}^2}/{4\pi^2} + {5x_1}/{\pi} - 6\right)^2 + 10\left(1 - {1}/{8\pi}\right)\cos x_1 + 10 \qquad \text{(C.1)}$$

$x_1 \in [-5,10], \qquad x_2[0,15]$

3. Camelback function (CB)

$$y(\boldsymbol{x}) = \left(4 - 2.1{x_1}^2 + {{x_1}^4}/{3}\right)^2 + x_1x_2 + (-4 + 4{x_2}^2){x_2}^2 \qquad \text{(C.2)}$$

$x_1 \in [-3,3], \qquad x_2[-2,2]$

4. Goldstein-Price function (GP)

$$y(\boldsymbol{x}) = [1 + (x_1 + x_2 + 1)^2 \times (19 - 4x_1 + 3{x_1}^2 - 14x_2 + 6x_1x_2 + 3{x_2}^2)] \times [30 + (2x_1 - 3x_2)^2] \times (18 - 32x_1 + 12{x_1}^2 + 48x_2 - 36x_1x_2 + 27{x_2}^2) \qquad \text{(C.3)}$$

$x_1 \in [-2,2], \qquad x_2[-2,2]$

5. Hartman function (H)

$$y(\boldsymbol{x}) = -\sum_{i=1}^{m} c_i \, exp\left\{-\sum_{j=1}^{n} a_{ij}\left(x_j - p_{ij}\right)^2\right\} \qquad \text{(C.4)}$$

$x_i \in [-1,1]$

Two specific instances are selected relies on the number of design variables. The value of 'm' is pre-set to 4.

5.1. Hartman 3 (HM3)

This instance has three variables, and the exact value of parameters are given by (Chen and Wang, 2013), as shown in Table C.1.

Table C.1. Parameters in Hartman function with three variables

#	a_{ij}			c_i	p_{ij}		
1	3.0	10.0	30.0	1.0	0.3689	0.1170	0.2673
2	0.1	10.0	35.0	1.2	0.4699	0.4387	0.7470
3	3.0	10.0	30.0	3.0	0.1091	0.8732	0.5547
4	0.1	10.0	35.0	3.2	0.03815	0.5743	0.8828

5.2.Hartman 6 (HM6)

This instance has six variables, and the choice of parameters is given in (Chen and Wang, 2013).

Table C.2. Parameters in Hartman function with three variables

#	a_{ij}						c_i	p_{ij}					
1	10.0	3.0	17.0	3.5	1.7	8.0	1.0	0.1312	0.1696	0.5529	0.0124	0.8283	0.5886
2	0.05	10.0	17.0	0.1	8.0	14.0	1.2	0.2329	0.4135	0.4135	0.3736	0.1004	0.9991
3	3.0	3.5	1.7	10.0	17.0	8.0	3.0	0.2348	0.1451	0.1451	0.2883	0.3047	0.6650

4	17.0	8.0	0.05	10.0	0.1	14.0	3.2	0.4047	0.8828	0.8828	0.5741	0.1091	0.0382

1. T1

$$y(\boldsymbol{x}) = (x_1 - 1)^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 \quad x_i \in [-3,3] \qquad (C.5)$$

2. T2

$$y(\boldsymbol{x}) = {x_1}^2 + {x_2}^2 + x_1 x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 \; x_i \in [0,15] \quad (C.6)$$

3. T3

$$y(\boldsymbol{x}) = \sum_{i=1}^{5}\{(\ln(x_i - 2))^2 + (\ln(10 - x_i))^2\} - \prod_{i=1}^{5} {x_i}^2 \, , x_i \in [2.1,9.9] \qquad (C.7)$$

4. T4

$$y(\boldsymbol{x}) = \sum_{i=1}^{10}\{(\ln(x_i - 2))^2 + (\ln(10 - x_i))^2\} - \prod_{i=1}^{10} {x_i}^2 \, , x_i \in [2.1,9.9] \qquad (C.8)$$

5. T5

$$y(\boldsymbol{x}) = {x_1}^2 + {x_2}^2 + x_1 x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5{x_7}^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45 \quad x_i \in [0,15] \qquad (C.9)$$

6. Benchmark function with noise

Noise in this work is created from a normal distribution: $noise \sim N(0, \sigma^2)$. The mean is equal to 0 and the variance represented as σ^2 is equal to 1/100 of that of the noise-free part of each benchmark function.

7. Design of Experiments

We generate an extensive collection of different data sets from different benchmark functions using the Design of Experiments (DoE). We build a series of experiments to identify the appropriate type of DoE. Usually, the selection of the most appropriate DOE depends on the problem and the aim of the experimentation. Since our focus is not on selecting the best DoE, so we make our decision on DoE type just based on conventional rules. Latin hypercube method is selected because using it we are able to reduce the number of the data and capture more information simultaneously (Cavazzuti, 2012). In addition, Kaufman and coauthors (Kaufman et al., 1996) propose that $3/4 * (dim + 1)(dim + 2)$ sample points are feasible for a problem with 5-10 variables, which shows that $(dim + 1)(dim + 2)$ sample points guarantee the rationality of the data to a large extent. As such, the equation that authors use to generate the number of data is presumed to be described by

$$N = m * (dim + 1)(dim + 2) \quad \text{(C.10)}$$

where N denotes the number of samples, dim denotes the number of input variables, m is a sequence from 1 to 10. Also, the comparison between surrogate models can be realized as m changes from 1 to 10 along with the gradually increased samples.

Appendix D. Sample data for the hot rod rolling problem (Section 6.4.1)

Appendix D contains the sample points for the three objectives, allotriomorphic ferrite (Y1), Widmanstätten ferrite (Y2) and pearlite (Y3). To generate test data to validate the performance of these surrogate models, we repeat a 9-fold cross-validation process 10 times. In each run, all data sets are randomly partitioned into 9 subsamples (groups). Of the 9 subsamples, one subsample is used as the testing data set and the remaining eight subsamples are used to train the model. Through 9 repetitions, all observations are involved in training and testing. Three output variables, namely, allotriomorphic ferrite (Y1), Widmanstätten ferrite (Y2) and pearlite (Y3) of steel are predicted

based on values of five input variables, namely, carbon concentration rate, manganese concentration rate, grain size, cooling rate, and final temperature. The prediction results are shown in Table D1.

Table D1. Values for input and output variables

Input					**Output**		
Concentrations (%)		Grain Size (μm)	Cooling Rate (K/min)	Final Temp (K)	Volume fractions (%)		
Carbon	Manganese				ALLOTRIOMORPHIC	WIDMANSTATTEN	PEARLITE
0.18	0.7	30	11	639.5	0.6921	0.0195	0.2883
0.24	0.7	55	11	643.2	0.4612	0.1621	0.3761
0.3	0.7	100	11	643.7	0.2429	0.2705	0.4866
0.24	0.7	30	55	620.9	0.3453	0.2352	0.4195
0.3	0.7	55	55	612.3	0.2439	0.2442	0.5113
0.18	0.7	100	55	624.8	0.2094	0.4899	0.2991
0.3	0.7	30	100	601.4	0.3163	0.1155	0.5681
0.18	0.7	55	100	612.4	0.2955	0.3944	0.3101
0.24	0.7	100	100	598.2	0.1458	0.4292	0.4243
0.24	1.1	30	11	625.1	0.5684	0.019	0.4126
0.3	1.1	55	11	626.7	0.3384	0.1439	0.5176
0.18	1.1	100	11	633.7	0.3132	0.3937	0.2915
0.3	1.1	30	55	599.3	0.3316	0.0888	0.5976
0.18	1.1	55	55	608.7	0.3179	0.3714	0.3107
0.24	1.1	100	55	600.9	0.1016	0.4618	0.4366
0.18	1.1	30	100	596.9	0.4185	0.2573	0.3242
0.24	1.1	55	100	587.6	0.2197	0.3122	0.4667
0.3	1.1	100	100	572.4	0.1042	0.2828	0.613
0.3	1.5	30	11	615.9	0.4575	0	0.5425
0.3	1.5	55	11	611.9	0.3085	0.14	0.5515
0.24	1.5	100	11	614.1	0.2367	0.3321	0.4312
0.18	1.5	30	55	592.7	0.3148	0.3523	0.3329
0.24	1.5	55	55	584.7	0.2129	0.3236	0.4618
0.3	1.5	100	55	572.2	0.1106	0.2527	0.6367
0.24	1.5	30	100	567.7	0.3197	0.185	0.4953
0.3	1.5	55	100	556.6	0.1586	0.162	0.6794
0.18	1.5	100	100	568.2	0.1434	0.4933	0.3102

Appendix E. How to Create a Variance-Covariance Matrix

Suppose **X** is an *n* x *k* matrix holding ordered sets of raw data. For example, matrix **X** might display the scores on *k* tests for *n* students. Starting with the raw data of matrix **X**, you can create a variance-covariance matrix to show the variance within each column and the covariance between columns. Here's how:

- Transform the raw scores from matrix **X** into deviation scores for matrix **x**.

$$x = X - 11'X (1 / n) \qquad (E.1)$$

where 1 is an *n* x *1* column vector of ones 1' is the transpose of vector 1

x is an *n* x *k* matrix of *deviation* scores: $x_{11}, x_{12}, \ldots, x_{nk}$

X is an *n* x *k* matrix of *raw* scores: $X_{11}, X_{12}, \ldots, X_{nk}$

- Compute x'x, the *k* x *k* deviation sums of squares and cross products matrix for x.

- Then, divide each term in the deviation sums of squares and cross product matrix by *n* to create the variance-covariance matrix. That is,

$$V = x'x (1 / n) \qquad (E.2)$$

Where ***V*** is a *k* * *k* variance-covariance matrix ***x'*** is the transpose of matrix *x*. ***x'x*** is the deviation sums of squares and cross product matrix *n* is the number of scores in each column of the original matrix ***X*.**

Example: The table below displays scores on math, English, and art tests for 5 students.

Student	Math	English	Art
1	90	60	90
2	90	90	30

3	60	60	60
4	60	60	90
5	30	30	30

Note that data from the table can be represented in matrix **A**, where each column in the matrix shows scores on a test and each row shows scores for a student.

$$\mathbf{A} = \begin{bmatrix} 90 & 60 & 90 \\ 90 & 90 & 30 \\ 60 & 60 & 60 \\ 60 & 60 & 90 \\ 30 & 30 & 30 \end{bmatrix}$$

Given the data represented in matrix **A**, compute the variance of each test and the covariance between the tests.

Solution

The solution involves a three-step process.

- First, we transform the *raw* scores in matrix **A** to *deviation* scores in matrix **a**, using the transformation formula described below:

Appendix F. How to Compute Deviation Scores (Section 8.4.3)

This lesson explains how to use matrix methods to transform raw scores to deviation scores. We show the transformation to deviation scores for vectors and for matrices.

Deviation Scores: Vectors

A deviation score is the difference between a raw score and the mean.

$$\boldsymbol{d_i} = \boldsymbol{x_i} - \boldsymbol{\bar{x}} \qquad (F.1)$$

Where d_i is the deviation score for the i^{th} observation in a set of observations

x_i is the raw score for the i^{th} observation in a set of observations

$\bar{x}$ is the mean of all the observations in a set of observations

Often, it is easier to work with deviation scores than with raw scores. Use the following formula to transform a vector of *n* raw scores into a vector of *n* deviation scores.

$$d = x - 1'x1\ (\ 1'1\ )^{-1} = x - 1'x1\ (\ 1/n\ ) \qquad (F.2)$$

where **1** is an *n* x *1* column vector of ones

d is an *n* x *1* column vector of *deviation* scores: $d_1, d_2, \ldots, d_n$

x is an *n* x *1* column vector of *raw* scores: $x_1, x_2, \ldots, x_n$

To show how this works, let's transform the raw scores in vector **x** to deviation scores in vector **d**. For this example, let **x**' = [1 2 3].

$$\mathbf{d} = \mathbf{x} - \mathbf{1}'\mathbf{x}\mathbf{1}\ (\ \mathbf{1}'\mathbf{1}\ )^{-1}$$

$$\mathbf{d} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} - [\,1\ 1\,1\,] \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} (\ [\,1\ 1\,1\,] \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}\ )^{-1}$$

$$\mathbf{d} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} - \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

Note that the mean deviation score is zero.

Deviation Scores: Matrices

Let **X** be an *r* x *c* matrix holding *raw* scores; and let **x** be the corresponding *r* x *c* matrix holding *deviation* scores. When transforming raw scores from **X** into deviation scores for **x**, we often want to compute deviation scores separately *within* columns, consistent with the equation below.

$$\boldsymbol{x_{rc}} = \boldsymbol{X_{rc}} - \overline{\boldsymbol{X_c}} \qquad (F.3)$$

Where $\boldsymbol{x_{rc}}$ is the deviation score from row *r* and column *c* of matrix **x**

$\boldsymbol{X_{rc}}$ is the raw score from row *r* and column *c* of matrix **X**

$\overline{\boldsymbol{X_c}}$ is the mean score, based on all *r* scores from column *c* of matrix X.

To transform the raw scores from matrix **X** into deviation scores for matrix **x**, we use this matrix equation.

$$x = X - 11'X (1'1)^{-1} = X - 11'X (1 / r) \qquad (F.4)$$

where 1 is an *r* x *1* column vector of ones

x is an *r* x *c* matrix of *deviation* scores: $x_{11}, x_{12}, \ldots, x_{rc}$

X is an *r* x *c* matrix of *raw* scores: $X_{11}, X_{12}, \ldots, X_{rc}$

Note: Deviation score matrices are often denoted by a lower-case, boldface letter, such as **x**. This can cause confusion, since vectors are also denoted by lower-case, boldface letters; but usually the meaning is clear from the context.

Example: Consider matrix X.

$$X = \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix}$$

Using matrix methods, create a 2 x 3 vector D, such that the elements of D are deviation scores based on elements from X. That is,

$$D = \begin{bmatrix} 3 - X_1 & 5 - X_2 & 1 - X_3 \\ 9 - X_1 & 1 - X_2 & 4 - X_3 \end{bmatrix}$$

where X_c is the mean of elements from column c of matrix X.

Solution

To solve this problem, we use the following equation: D = X - 11'X (1 / r). Each step in the computation is shown below.

$$D = X - 1 \quad 1' \quad X \quad (1/r)$$

$$D = \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \end{bmatrix} [\,1\ 1\,] \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix} (1/r)$$

$$D = \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix} (1/2)$$

$$D = \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 12 & 6 & 5 \\ 12 & 6 & 5 \end{bmatrix} (1/2)$$

$$D = \begin{bmatrix} 3 & 5 & 1 \\ 9 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 6 & 3 & 2.5 \\ 6 & 3 & 2.5 \end{bmatrix}$$

$$D = \begin{bmatrix} -3 & 2 & -1.5 \\ 3 & -2 & 1.5 \end{bmatrix}$$

Thus, matrix D has the deviation scores, based on raw scores from matrix X. Note that the mean and sum of each column in matrix D is zero.

Reference

Aaltonen M, Oksanen A, Kivivuori J. Debt problems and crime. Criminology 2016;54; 307-331.

Abudu S, King JP, Bawazir AS. Forecasting monthly streamflow of spring-summer runoff season in rio grande headwaters basin using stochastic hybrid modeling approach. Journal of Hydrologic Engineering 2010;16; 384-390.

Acar E. Effect of error metrics on optimum weight factor selection for ensemble of metamodels. Expert Systems with Applications 2015;42; 2703-2709.

Acar E, Rais-Rohani M. Ensemble of metamodels with optimized weight factors. Structural and Multidisciplinary Optimization 2009;37; 279-294.

Adamowski JF. Peak daily water demand forecast modeling using artificial neural networks. Journal of Water Resources Planning and Management 2008;134; 119-128.

Adhav R, Samad A, Kenyery F, 2015. Design optimization of electric centrifugal pump by multiple surrogate models, SPE Middle East Oil & Gas Show and Conference. Society of Petroleum Engineers, Manama, Bahrain.

Afshari H, Sharafi M, ElMekkawy T, Peng Q. Optimizing multi-objective dynamic facility location decisions within green distribution network design. Procedia CIRP 2014;17; 675-679.

Alizadeh R, Allen JK, Mistree F. Managing computational complexity using surrogate models: a critical review. Research in Engineering Design 2020a;31; 275-298.

Alizadeh R, Gharizadeh Beiragh R, Soltanisehat L, Soltanzadeh E, Lund PD. Performance evaluation of complex electricity generation systems: A dynamic network-based data envelopment analysis approach. Energy Economics 2020b;91; 104894.

Alizadeh R, Jia L, Nellippallila AB, Wang G, Hao J, Allen JK, Mistree F. Ensemble of surrogates and cross-validation for rapid and accurate predictions using small data sets. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2019;33; 484-501.

Alizadeh R, Khodaei R, Maknoon R. A combined model of scenario planning and assumption-based planning for futurology, and robust decision making in the energy sector. Quarterly Journal Of Energy Policy And Planning Research 2016a;2; 7-32.

Alizadeh R, Lund PD, Beynaghi A, Abolghasemi M, Maknoon R. An integrated scenario-based robust planning approach for foresight and strategic management with application to energy industry. Technological Forecasting and Social Change 2016b;104; 162-171.

Alizadeh R, Lund PD, Soltanisehat L. Outlook on biofuels in future studies: A systematic literature review. Renewable and Sustainable Energy Reviews 2020c;134; 110326.

Alizadeh R, Majidpour M, Maknoon R, Kaleibari SS. Clean development mechanism in Iran: does it need a revival? International Journal of Global Warming 2016c;10; 196-215.

Alizadeh R, Majidpour M, Maknoon R, Salimi J. Iranian energy and climate policies adaptation to the Kyoto protocol. International Journal of Environmental Research 2015a;9; 853-864.

Alizadeh R, Maknoon R, Majidpour M, 2014. Clean development mechanism, a bridge to mitigate the greenhouse gases: is it broke in Iran?, 13th International Conference on Clean Energy (ICCE-2014), Istanbul, Turkey, pp. 399-404.

Alizadeh R, Maknoon R, Majidpour M, Salimi J. Energy policy in iran and international commitments for GHG emission reduction. Journal of Environmental Science and Technology 2015b;17; 183-198.

Alizadeh R, Soltanisehat L. Stay competitive in 2035: A scenario-based method to foresight in the design and manufacturing industry. Foresight 2020.

Alizadeh R, Soltanisehat L, Lund PD, Zamanisabzi H. Improving renewable energy policy planning and decision-making through a hybrid MCDM method. Energy Policy 2020d;137; 111174.

Almanie T, Mirza R, Lor E, 2015. Crime prediction based on crime types and using spatial and temporal criminal hotspots, ArXiv E-prints.

Altmann M. A supply chain design approach considering environmentally sensitive customers: the case of a German manufacturing small and medium enterprises. International Journal of Production Research 2015;53; 6534-6550.

Amit Y, Geman D. Shape quantization and recognition with randomized trees. Neural Computation 1997;9; 1545-1588.

Anderson JM, MacDonald JM, Bluthenthal R, Ashwood JS. Reducing crime by shaping the built environment with zoning: an empirical study of los angeles. University of Pennsylvania Law Review 2013;161; 699-756.

Andresen MA. Environmental criminology: evolution, theory, and practice. Routledge: NY; 2019.

Anselin L. Spatial econometrics: methods and models. Springer Science & Business Media: NY; 2013.

Applegate D, Cook W, Rohe A. Chained Lin-Kernighan for large traveling salesman problems. INFORMS Journal on Computing 2003;15; 82-92.

Arias-Montano A, Coello CAC, Mezura-Montes E, 2012a. Multi-objective airfoil shape optimization using a multiple-surrogate approach, Evolutionary Computation (CEC), 2012 IEEE Congress on. IEEE, NY, pp. 1-8.

Arias-Montano A, Coello CAC, Mezura-Montes E, 2012b. Multi-objective airfoil shape optimization using a multiple-surrogate approach, IEEE Congress on Evolutionary Computation (CEC). IEEE, Brisbane, Australia, pp. 1-8.

Atsalakis G, Minoudaki C, Markatos N, Stamou A, Beltrao J, Panagopoulos T, 2007. Daily irrigation water demand prediction using adaptive neuro-fuzzy inferences systems (anfis), Proceedings of International Conference on Energy, Environment, Ecosystems and Sustainable Development, Agios Nikolaos, Greece, Island.

Babaei M, Pan I. Performance comparison of several response surface surrogate models and ensemble methods for water injection optimization under uncertainty. Computers & Geosciences 2016;91; 19-32.

Badhurshah R, Samad A. Multiple surrogate based optimization of a bidirectional impulse turbine for wave energy conversion. Renewable Energy 2015;74; 749-760.

Baller RD, Anselin L, Messner SF, Deane G, Hawkins DF. Structural covariates of US county homicide rates: Incorporating spatial effects. Criminology 2001;39; 561-588.

Balling R, Wilkinson C. Execution of multidisciplinary design optimization approaches on common test problems. AIAA Journal 1997;35; 178-186.

Bandler JW, Madsen K. Surrogate modelling and space mapping for engineering optimization. Optimization and Engineering 2001;2; 367-368.

Barnes G, Langworthy P, 2003. The per-mile costs of operating automobiles and trucks. University of Minnesota, Humphrey Institute of Public Affairs, Minnesota, USA, pp. 1-46.

Barthelemy J-FM, Haftka RT. Approximation concepts for optimum structural design — a review. Structural optimization 1993;5; 129-144.

Basudhar A, 2012. Selection of anisotropic kernel parameters using multiple surrogate information, 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. AIAA, Michigan, p. 5576.

Beck JL, Au S-K. Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation. Journal of Engineering Mechanics 2002;128; 380-391.

Beiragh RG, Alizadeh R, Kaleibari SS, Cavallaro F, Zolfani SH, Bausys R, Mardani A. An integrated multi-criteria decision making model for sustainability performance assessment for insurance companies. Sustainability 2020;12; 789.

Bektas T. The multiple traveling salesman problem: an overview of formulations and solution procedures. Omega 2006;34; 209-219.

Bektaş T, Demir E, Laporte G, 2016. Green vehicle routing, Green Transportation Logistics. Springer; 2016. pp. 243-265.

Beland L-P, Brent DA. Traffic and crime. Journal of Public Economics 2018;160; 96-116.

Bellary SAI, Adhav R, Siddique MH, Chon B-H, Kenyery F, Samad A. Application of computational fluid dynamics and surrogate-coupled evolutionary computing to enhance centrifugal-pump performance. Engineering Applications of Computational Fluid Mechanics 2016;10; 171-181.

Bellary SAI, Samad A. An alternative approach to surrogate averaging for a centrifugal impeller shape optimisation. International Journal of Computer Aided Engineering and Technology 2017;9; 62-83.

Bellucci JP, Bauer Jr KW. A Taylor series approach to the robust parameter design of computer simulations using kriging and radial basis function neural networks. International Journal of Quality Engineering and Technology 2017;6; 137-160.

Benjaafar S, Li Y, Daskin M. Carbon footprint and the management of supply chains: Insights from simple models. IEEE Transactions on Automation Science and Engineering 2012;10; 99-116.

Bennett C, Stewart R, Lu J. Autoregressive with exogenous variables and neural network short-term load forecast models for residential low voltage distribution networks. Energies 2014;7; 2938-2960.

Bernasco W, Elffers H, 2010. Statistical analysis of spatial crime data, Handbook of Quantitative Criminology. Springer, Germany; 2010. pp. 699-724.

Bertazzi L, Secomandi N. Faster rollout search for the vehicle routing problem with stochastic demands and restocking. European Journal of Operational Research 2018;270; 487-497.

Bertsimas DJ. A vehicle routing problem with stochastic demand. Operations Research 1992;40; 574-585.

Bettebghor D, Bartoli N, Grihon S, Morlier J, Samuelides M. Surrogate modeling approximation using a mixture of experts based on EM joint estimation. Structural and Multidisciplinary Optimization 2011;43; 243-259.

Beynaghi A, Moztarzadeh F, Shahmardan A, Alizadeh R, Salimi J, Mozafari M. Makespan minimization for batching work and rework process on a single facility with an aging effect: a hybrid meta-heuristic algorithm for sustainable production management. Journal of Intelligent Manufacturing 2016;30; 33-45.

Bhat S, Viana FAC, Lind R, Haftka R, 2010. Control-oriented design using H-infinity synthesis and multiple surrogates, 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference. AIAA, Orlando, Florida, p. 3089.

Bhattacharjee KS, Isaacs A, Ray T, 2017. Multi-objective optimization using an evolutionary algorithm embedded with multiple spatially distributed surrogates, MULTI-OBJECTIVE OPTIMIZATION: Techniques and Application in Chemical Engineering. World Scientific, NY; 2017. pp. 135-155.

Bhattacharjee KS, Singh HK, Ray T, Branke J, 2016. Multiple surrogate assisted multiobjective optimization using improved pre-selection, 2016 IEEE Congress on Evolutionary Computation (CEC). IEEE, Orlando, Florida, pp. 4328-4335.

Bhattacharjee KS, Singh HK, Tapabrata R. Multiple surrogate assisted many-objective optimization for computationally expensive engineering design. ASME Journal of Mechanical Design 2018;140; 051403-051401-051403-051410.

Bhingea R, Moser R, Moser E, Lanzab G, Dornfelda D. Sustainability optimization for global supply chain decision-making. Procedia CIRP 2015;26; 323 – 328.

Bishop CM. Neural networks for pattern recognition. Oxford University Press: Department of Computer Science and Applied Mathematics Aston University Birmingham, UK; 1995.

Bliznyuk N, Ruppert D, Shoemaker C, Regis R, Wild S, Mugunthan P. Bayesian calibration and uncertainty analysis for computationally expensive models using optimization and radial basis function approximation. Journal of Computational and Graphical Statistics 2008;17; 270-294.

Bock C, Zha X, Suh H-w, Lee J-H. Ontological product modeling for collaborative design. Advanced Engineering Informatics 2010;24; 510-524.

Bock S, Klamroth K. Combining traveling salesman and traveling repairman problems: a multi-objective approach based on multiple scenarios. Computers & Operations Research 2019;112; 104766.

Bodnar R, Hansen S. Effects of austenite grain size and cooling rate on Widmanstätten ferrite formation in low-alloy steels. Metallurgical and Materials Transactions A 1994;25; 665-675.

Bouma HW, Teunter RH. The routed inventory pooling problem with multiple lateral transshipments. International Journal of Production Research 2016;54; 3523-3533.

Box GE, Draper NR. Empirical model-building and response surfaces. John Wiley & Sons: NY; 1987.

Box GE, Hunter JS. The 2 k—p fractional factorial designs. Technometrics 1961;3; 311-351.

Box GE, Hunter WG, Hunter JS. Statistics for experimenters. John Wiley and Sons: New York, United States; 1978a.

Box GE, Jenkins GM, Reinsel GC, Ljung GM. Time series analysis: forecasting and control. John Wiley and Sons: Hoboken, NJ; 2015.

Box GEP, Behnken DW. Some New Three Level Designs for the Study of Quantitative Variables. Technometrics 1960;2; 455-475.

Box GEP, Hunter JS. Multi-factor experimental designs for exploring response surfaces. The Annals of Mathematical Statistics 1957;28; 195-241.

Box GEP, Hunter WG, Hunter JS. Statistics for experimenters. John Wiley & Sons: NY; 1978b.

Braekers K, Ramaekers K, Van Nieuwenhuyse I. The vehicle routing problem: State of the art classification and review. Computers & Industrial Engineering 2016;99; 300-313.

Bramel J, Simchi-Levi D. The logic of logistics: theory, algorithms, and applications for logistics management. Springer; 1997.

Brandenburg M, Govindan K, Sarkis J, Seuring S. Quantitative models for sustainable supply chain management: Developments and directions. European Journal of Operational Research 2014;233; 299-312.

Breiman L. Random forests. Machine Learning 2001;45; 5-32.

Brereton RG, Lloyd GR. Support vector machines for classification and regression. Analyst 2010;135; 230-267.

Brimberg J, Hansen P, Mladenović N, Taillard ED. Improvements and comparison of heuristics for solving the uncapacitated multisource Weber problem. Operations Research 2000;48; 444-460.

Brockwell PJ, Davis RA. Time series: theory and methods. Springer Science & Business Media: NY, USA; 2009.

Broomhead DS, Lowe D. Multivariable functional interpolation and adaptive networks. J Complex Systems 1988;2; 321--355.

Burley BA. Green infrastructure and violence: do new street trees mitigate violent crime? Health & Place 2018;54; 43-49.

Burnham KP, Anderson DR. Model selection and multimodel inference: a practical information-theoretic approach. Springer Science & Business Media: Germany; 2003.

Cachon G. Managing a retailer's shelf space, inventory, and transportation. Manufacturing & Service Operations Management 2001;3; 211-229.

Cachon GP. Retail store density and the cost of greenhouse gas emissions. Management Science 2014;60; 1907-1925.

Campelo P, Neves-Moreira F, Amorim P, Almada-Lobo B. Consistent vehicle routing problem with service level agreements: A case study in the pharmaceutical distribution sector. European Journal of Operational Research 2019;273; 131-145.

Canter D, Shalev K, 2017. Putting crime in its place: psychological process in crime site selection, Principles of Geographical Offender Profiling. Routledge, New York, United States; 2017. pp. 275-286.

Carter AE, Ragsdale CT. A new approach to solving the multiple traveling salesperson problem using genetic algorithms. European Journal of Operational Research 2006;175; 246-257.

Cavazzuti M, 2012. Design of experiments, Optimization Methods: From Theory to Design Scientific and Technological Aspects in Mechanics. Springer Science & Business Media, Berlin; 2012. pp. 13-22.

Chatterjee S, Carrera C, Lynch LA. Genetic algorithms and traveling salesman problems. European Journal of Operational Research 1996;93; 490-510.

Chaudhuri A, Haftka RT. Efficient global optimization with adaptive target setting. AIAA Journal 2014;52; 1573-1578.

Chaudhuri A, Haftka RT, Chang K, Van Hall J, Ifju P, 2014. Thrust-power pareto fronts based on experiments for small flapping wings, 10th AIAA Multidisciplinary Design Optimization Conference. American Institute of Aeronautics and Astronautics, New York, United States; 2014.

Chaudhuri A, Haftka RT, Ifju P, Chang K, Tyler C, Schmitz T. Experimental flapping wing optimization and uncertainty quantification using limited samples. Structural and Multidisciplinary Optimization 2015;51; 957-970.

Chauvin Y, Rumelhart DE. Backpropagation: theory, architectures, and applications. Psychology Press: New York, United States; 2013.

Chen B, Liu J, Zhong Y, Zhou J, 2000. Properties of the coupled factors in MDO and their application in collaborative optimization, Proceedings of the DETC'00, ASME 2000 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Baltimore, MD, Sept, pp. 10-13.

Chen J, Lin DK. On the identifiability of a supersaturated design. Journal of Statistical Planning and Inference 1998;72; 99-107.

Chen S, Wang X. A derivative-free optimization algorithm using sparse grid integration. American Journal of Computational Mathematics 2013;3; 16-26.

Chen VC, Tsui K-L, Barton RR, Allen JK. A review of design and modeling in computer experiments. Handbook of Statistics 2003;22; 231-261.

Chen VC, Tsui K-L, Barton RR, Meckesheimer M. A review on design, modeling and applications of computer experiments. IIE Transactions 2006;38; 273-291.

Chen X, Benjaafar S, Elomri A. The carbon-constrained EOQ. Operations Research Letters 2013;41; 172-179.

Cheng B, Titterington DM. Neural networks: a review from a statistical perspective. Statistical Science 1994;9; 2-54.

Cheng RC, Currie CS, 2004. Optimization by simulation metamodelling methods, Proceedings of the 36th Conference on Winter simulation. Winter Simulation Conference, New York, United States, pp. 485-490.

Chiang W-C, Li Y, Shang J, Urban TL. Impact of drone delivery on sustainability and cost: Realizing the UAV potential through vehicle routing optimization. Applied Energy 2019;242; 1164-1175.

Chitsaz M, Divsalar A, Vansteenwegen P. A two-phase algorithm for the cyclic inventory routing problem. European Journal of Operational Research 2016;254; 410-426.

Cho H, Bae S, Choi KK, Lamb D, Yang R-J. An efficient variable screening method for effective surrogate models for reliability-based design optimization. Structural and Multidisciplinary Optimization 2014;50; 717-738.

Choudhary A, Sarkar S, Settur S, Tiwari MK. A carbon market sensitive optimization model for integrated forward–reverse logistics. International Journal of Production Economics 2015;164; 433-444.

Cicirelli F, Fortino G, Guerrieri A, Spezzano G, Vinci A. Metamodeling of smart environments: from design to implementation. Advanced Engineering Informatics 2017;33; 274-284.

Clark PJ, Evans FC. Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations. Ecology 1954;35; 445-453.

Cohen J, Gorr WL. Development of crime forecasting and mapping systems for use by police. H. John Heinz III School of Public Policy and Management, Carnegie Mellon: New York, United States; 2005.

Cohen J, Gorr WL, Olligschlaeger AM. Leading indicators and spatial interactions: A crime-forecasting model for proactive police deployment. Geographical Analysis 2007;39; 105-127.

Colicchia C, Creazza A, Dallari F, Melacini M. Eco-efficient supply chain networks: development of a design framework and application to a real case study. Production Planning & Control 2016;27; 157-168.

Conley TG, Topa G. Socio-economic distance and spatial patterns in unemployment. Journal of Applied Econometrics 2002;17; 303-327.

Corchado E, Corchado E, Corchado JM, Abraham A. Innovations in Hybrid Intelligent Systems (Advances in Soft Computing). Springer Publishing Company, Incorporated: Berlin, Germany; 2007.

Cornish DB, Clarke RV. The reasoning criminal: Rational choice perspectives on offending. Transaction Publishers: NY, USA; 2014.

Correia I, Melo T. Multi-period capacitated facility location under delayed demand satisfaction. European Journal of Operational Research 2016;255; 729-746.

Couckuyt I, De Turck F, Dhaene T, Gorissen D, 2011. Automatic surrogate model type selection during the optimization of expensive black-box problems, Proceedings of the Winter Simulation Conference. Winter Simulation Conference, Phoenix, AZ, USA, pp. 4274-4284.

Coulibaly P, Anctil F, Bobee B. Daily reservoir inflow forecasting using artificial neural networks with stopped training approach. Journal of Hydrology 2000;230; 244-257.

Crama Y, Rezaei M, Savelsbergh M, Woensel TV. Stochastic inventory routing for perishable products. Transportation Science 2018;52; 526-546.

Cui T, Ouyang Y, Shen Z-JM. Reliable facility location design under the risk of disruptions. Operations Research 2010;58; 998-1011.

Cui Y, Lee J, Walla NJ, Silaen AK, Goodloe DD, Zhou CQ, 2018. Analysis of flow and thermal stress in a blast furnace blowpipe, ASME 2018 International Mechanical Engineering Congress and Exposition. ASME, Pittsburgh, Pennsylvania, USA, pp. 1-8.

Cutler A, Cutler DR, Stevens JR, 2012. Random forests, Ensemble machine learning. Springer, Germany; 2012. pp. 157-175.

Daganzo CF. The distance traveled to visit N points with a maximum of C stops per vehicle: An analytic model and an application. Transportation Science 1984;18; 331-350.

Dandge S, Chakraborty S, 2020. Selection of machining parameters in ultrasonic machining process using cart algorithm, Advanced Engineering Optimization Through Intelligent Techniques. Springer, Germany; 2020. pp. 599-607.

Dantzig GB, Ramser JH. The truck dispatching problem. Management Science 1959;6; 80-91.

Das K, Rao Posinasetti N. Addressing environmental concerns in closed loop supply chain design and planning. International Journal of Production Economics 2015;163; 34-47.

Daskin MS. Network and discrete location: models, algorithms, and applications. John Wiley & Sons: New York, USA; 2011.

De'Ath G. Boosted trees for ecological modeling and prediction. Ecology 2007;88; 243-251.

De Koster R, Le-Duc T, Roodbergen KJ. Design and control of warehouse order picking: A literature review. European Journal of Operational Research 2007;182; 481-501.

Deaton JD, Grandhi RV. A survey of structural and multidisciplinary continuum topology optimization: post 2000. Structural and Multidisciplinary Optimization 2014;49; 1-38.

Deb K, Hussein R, Roy P, Toscano G, 2017. Classifying metamodeling methods for evolutionary multi-objective optimization: first results, In: Trautmann H, Rudolph G, Klamroth K, Schütze O, Wiecek M, Jin Y, Grimme C (Eds), Evolutionary Multi-Criterion Optimization: 9th International Conference, EMO 2017, Münster, Germany, March 19-22, 2017, Proceedings. Springer International Publishing, Cham; 2017. pp. 160-175.

Deb K, Myburgh C, 2016. Breaking the billion-variable barrier in real-world optimization using a customized evolutionary algorithm, Proceedings of the 2016 on Genetic and Evolutionary Computation Conference. ACM, Denver, USA, pp. 653-660.

Defryn C, Sörensen K. A fast two-level variable neighborhood search for the clustered vehicle routing problem. Computers & Operations Research 2017;83; 78-94.

Demuth HB, Beale MH, De Jess O, Hagan MT. Neural network design. Pws Pub: Boston, United States; 2014.

Denkena B, Bergmann B, Witt M. Material identification based on machine-learning algorithms for hybrid workpieces during cylindrical operations. Journal of Intelligent Manufacturing 2018.

Dey A. Orthogonal fractional factorial designs. Wiley: NY, USA; 1985.

Di Francescomarino C, Dumas M, Federici M, Ghidini C, Maggi FM, Rizzi W, Simonetto L. Genetic algorithms for hyperparameter optimization in predictive business process monitoring. Information Systems 2018;74; 67-83.

Díaz-Manríquez A, Toscano G, Coello CAC. Comparison of metamodeling techniques in evolutionary algorithms. Soft Computing 2017;21; 5647-5663.

Dobler CP, Anderson-Cook CM. Forecasting, Time Series, and Regression: An Applied Approach. The American Statistician 2005;59; 278.

Draper NR, Guttman I. An index of rotatability. Technometrics 1988;30; 105-111.

Du X. Time-dependent mechanism reliability analysis with envelope functions and first-order approximation. Journal of Mechanical Design 2014;136; 081010.

Duda RO, Hart PE, Stork DG. Pattern classification. John Wiley & Sons: NY, USA; 2012.

Dumais S, Platt J, Heckerman D, Sahami M, 1998. Inductive learning algorithms and representations for text categorization, Proceedings of the Seventh International Conference on Information and Knowledge Management, Bethesda, Maryland, USA pp. 148-155.

Durbin R, Willshaw D. An analogue approach to the travelling salesman problem using an elastic net method. Nature 1987;326; 689-691.

Easterling RG, Berger JO, 2002. Statistical Foundations for the Validation of Computer Models, Presented at Computer Model Verification and Validation in the 21st Century Workshop. Johns Hopkins University, Baltimore, Maryland.

Eisaman MD, Rivest JLB, Karnitz SD, de Lannoy C-F, Jose A, DeVaul RW, Hannun K. Indirect ocean capture of atmospheric CO2: Part II. Understanding the cost of negative emissions. International Journal of Greenhouse Gas Control 2018;70; 254-261.

El-Shafie A, Taha MR, Noureldin A. A neuro-fuzzy model for inflow forecasting of the Nile River at Aswan High dam. Water Resources Management 2007;21; 533-556.

Elhedhli S, Merrick R. Green supply chain network design to reduce carbon emissions. Transportation Research Part D: Transport and Environment 2012;17; 370-379.

EPA, 2010. Managing supply chain greenhouse gas emissions :lessons learned for the road ahead. United States EPA, NY, USA.

EPA, 2019. Inventory of US greenhouse gas emissions and sinks: 1990–2019. US Environmental Protection Agency, NY, USA, pp. 21-32.

Erdoğan G. An open source Spreadsheet Solver for Vehicle Routing Problems. Computers & Operations Research 2017;84; 62-72.

Ezhilsabareesh K, Rhee SH, Samad A. Shape optimization of a bidirectional impulse turbine via surrogate models. Engineering Applications of Computational Fluid Mechanics 2018;12; 1-12.

Fahimnia B, Sarkis J, Davarzani H. Green supply chain management: A review and bibliometric analysis. International Journal of Production Economics 2015;162; 101-114.

Fang J, Sun G, Qiu N, Kim NH, Li Q. On design optimization for structural crashworthiness and its state of the art. Structural and Multidisciplinary Optimization 2017;55; 1091-1119.

Faust K, Tita GE. Social networks and crime: Pitfalls and promises for advancing the field. Annual Review of Criminology 2019;2; 99-122.

Federgruen A, Zipkin P. A combined vehicle routing and inventory allocation problem. Operations Research 1984;32; 1019-1037.

Fekete SP, Mitchell JS, Beurer K. On the continuous Fermat-Weber problem. Operations Research 2005;53; 61-76.

Fischetti M, Salazar González JJ, Toth P. A branch-and-cut algorithm for the symmetric generalized traveling salesman problem. Operations Research 1997;45; 378-394.

Fitz T, Theiler M, Smarsly K. A metamodel for cyber-physical systems. Advanced Engineering Informatics 2019;41; 100930.

Fokkema JE, Land MJ, Coelho LC, Wortmann H, Huitema GB. A continuous-time supply-driven inventory-constrained routing problem. Omega 2020;92; 102151.

Forrester AIJ, Keane AJ. Recent advances in surrogate-based optimization. Progress in Aerospace Sciences 2009;45; 50-79.

Fowler JW, Kim S-H, Shunk DL. Design for customer responsiveness: Decision support system for push–pull supply chains with multiple demand fulfillment points. Decision Support Systems 2019;123; 113071.

Frazzon EM. Sustainability and Effectiveness in Global Logistic Systems. GITO mbH Verlag; 2009.

Friedman J, Hastie T, Tibshirani R. The elements of statistical learning. Springer series in statistics New York, USA; 2001.

Friedman JH. Multivariate adaptive regression splines. Annals of Statistics 1991;19; 1-67.

Friedman JH. Greedy function approximation: a gradient boosting machine. Annals of statistics 2001; 1189-1232.

Fry M, Owen D. Green Metropolis: Why Living Smaller, Living Closer, and Driving Less Are the Keys to Sustainability. 2013;53; 126-128.

Gaitan C, McPherson R, H Rosendahl D, B Stacy M, Kellogg W, Austin B, 2016. Statistically downscaled datasets for the Red River Basin, South Central U.S.A. U.S. Geological Survey, United States.

Gallego G, Simchi-Levi D. On the effectiveness of direct shipping strategy for the one-warehouse multi-retailer R-systems. Management Science 1990;36; 240-243.

Gano SE, Renaud JE, Martin JD, Simpson TW. Update strategies for kriging models used in variable fidelity optimization. Structural and Multidisciplinary Optimization 2006;32; 287-298.

Gao H, Breitkopf P, Coelho RF, Xiao M. Categorical structural optimization using discrete manifold learning approach and custom-built evolutionary operators. Structural and Multidisciplinary Optimization 2018.

Gao N, Ryan SM. Robust design of a closed-loop supply chain network for uncertain carbon regulations and random product flows. EURO Journal on Transportation and Logistics 2014;3; 5-34.

Garbo A, German BJ. Performance assessment of a cross-validation sampling strategy with active surrogate model selection. Structural and Multidisciplinary Optimization 2019;59; 2257-2272.

Garg K, Kannan D, Diabat A, Jha PC. A multi-criteria optimization approach to manage environmental issues in closed loop supply chain network design. Journal of Cleaner Production 2015;100; 297-314.

Gehlert A, Esswein W. Toward a formal research framework for ontological analyses. Advanced Engineering Informatics 2007;21; 119-131.

Geisser S. Predictive inference. CRC press: NY, USA; 1993.

Geist A. PVM-parallel virtual machine : a users' guide and tutorial for networked parallel computing. MIT Press: Cambridge, Mass.; 2000.

Ghanem RG, Doostan A. On the construction and analysis of stochastic models: characterization and propagation of the errors associated with limited data. Journal of Computational Physics 2006;217; 63-81.

Gharizadeh Beiragh R, Alizadeh R, Shafiei Kaleibari S, Cavallaro F, Hashemkhani Zolfani S, Bausys R, Mardani A. An integrated multi-criteria decision making model for sustainability performance assessment for insurance companies. Sustainability 2020;12; 789.

Giunta AA, Dudley JM, Narducci R, Grossman B, Haftka RT, Mason WH, and Watson LT, 1994. Noisy aerodynamic response and smooth approximations in HSCT design, 5th Symposium on Multidisciplinary Analysis and Optimization. American Institute of Aeronautics and Astronautics; 1994.

Glaeser EL. Triumph of the city: How our greatest invention makes us richer, smarter, greener, healthier, and happier (an excerpt). Journal of Economic Sociology 2013;14; 75-94.

Glaz B, Goel T, Liu L, Friedmann PP, Haftka RT. Multiple-surrogate approach to helicopter rotor blade vibration reduction. Aiaa Journal 2009;47; 271-282.

Goel T, Haftka RT, Shyy W, Queipo NV. Ensemble of surrogates. Structural and Multidisciplinary Optimization 2007;33; 199-216.

Gómez-Chova L, Calpe J, Soria E, Camps-Valls G, Martin J, Moreno J, 2003. CART-based feature selection of hyperspectral images for crop cover classification, Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429). IEEE, pp. III-589.

Goodchild FM. Scale in GIS: An overview. Geomorphology 2011;130; 5-9.

Gorissen D, Couckuyt I, Laermans E, Dhaene T. Multiobjective global surrogate modeling, dealing with the 5-percent problem. Engineering with Computers 2010;26; 81-98.

Gorr W, Harries R. Introduction to crime forecasting. International Journal of Forecasting 2003;19; 551-555.

Gottlieb A, Sugie NF. Marriage, cohabitation, and crime: Differentiating associations by partnership stage. Justice Quarterly 2019;36; 503-531.

Govindan K, Fattahi M, Keyvanshokooh E. Supply chain network design under uncertainty: A comprehensive review and future research directions. European Journal of Operational Research 2017;263; 108-141.

Govindan K, Jafarian A, Nourbakhsh V. Designing a sustainable supply chain network integrated with vehicle routing: A comparison of hybrid swarm intelligence metaheuristics. Computers & Operations Research 2019;110; 220-235.

Grama A. Introduction to parallel computing. Addison-Wesley: Harlow, England; New York; 2013.

Gray AC, Parker KF. Race, structural predictors, and police shootings: are there differences across official and "unofficial" accounts of lethal force? Crime & Delinquency 2019;65; 26-45.

Grewal D, Kopalle P, Marmorstein H, Roggeveen AL. Does travel time to stores matter? The role of merchandise availability. Journal of Retailing 2012;88; 437-444.

Gunn SR. Support vector machines for classification and regression. ISIS technical report 1998;14; 85-86.

Gunst RF, 1996. Response surface methodology: process and product optimization using designed experiments. Taylor & Francis Group.

Gunst RF, Mason RL. Fractional factorial design. Wiley Interdisciplinary Reviews: Computational Statistics 2009;1; 234-244.

Güntert P, Billeter M, Ohlenschläger O, Brown LR, Wüthrich K. Conformational analysis of protein and nucleic acid fragments with the new grid search algorithm FOUND. Journal of Biomolecular NMR 1998;12; 543-548.

Gürbüz MÇ, Moinzadeh K, Zhou Y-P. Coordinated replenishment strategies in inventory/distribution systems. Management Science 2007;53; 293-307.

Gustafsson A, Herrmann A, Huber F. Conjoint measurement: Methods and applications. Springer Science & Business Media; 2013.

Gutmann H-M. A radial basis function method for global optimization. Journal of global optimization 2001;19; 201-227.

Haberman CP. Overlapping Hot Spots? Criminology & Public Policy 2017;16; 633-660.

Habib A, Kumar Singh H, Ray T. A multiple surrogate assisted evolutionary algorithm for optimization involving iterative solvers. Engineering Optimization 2017; 1-20.

Haddadsisakht A, Ryan SM. Closed-loop supply chain network design with multiple transportation modes under stochastic demand and uncertain carbon tax. International Journal of Production Economics 2018;195; 118-131.

Haimovich M, Rinnooy Kan AH. Bounds and heuristics for capacitated routing problems. Mathematics of operations Research 1985;10; 527-542.

Hajihashemi S, Alizadeh R, Allen JK, Mistree F, 2021. Impact of asset management in a green supply chain, ASME 2021 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference.

Han B, Cohen DA, Derose KP, Li J, Williamson S. Violent crime and park use in low-income urban neighborhoods. American Journal of Preventive Medicine 2018;54; 352-358.

Han J, Pei J, Kamber M. Data mining: concepts and techniques. Elsevier; 2011.

Hansen PC, Pereyra V, Scherer G. Least squares data fitting with applications. JHU Press; 2012.

Hanson RF, Smith DW, Kilpatrick DG, Freedy JR. Crime-related fears and demographic diversity in los angeles county after the 1992 civil disturbances. Journal of Community Psychology 2000;28; 607-623.

Harris I, Mumford CL, Naim MM. A hybrid multi-objective approach to capacitated facility location with flexible store allocation for green logistics modeling. Transportation Research Part E: Logistics and Transportation Review 2014;66; 1-22.

Hasselman T, Yap K, Lin C, Cafeo J, 2005. A case study in model improvement for vehicle crashworthiness simulation, 23rd International Modal Analysis Conference.

He X, Wang L, Hong HG. Quantile-adaptive model-free variable screening for high-dimensional heterogeneous data. The Annals of Statistics 2013;41; 342-369.

Hedayat AS, Sloane NJA, Stufken J. Orthogonal arrays: theory and applications. Springer Science & Business Media; 2012.

Helbich M, Griffith DA. Spatially varying coefficient models in real estate: Eigenvector spatial filtering and alternative approaches. Computers, Environment and Urban Systems 2016;57; 1-11.

Helbich M, Jokar Arsanjani J. Spatial eigenvector filtering for spatiotemporal crime mapping and spatial crime analysis. Cartography and Geographic Information Science 2015;42; 134-148.

Helsgaun K. An effective implementation of the Lin–Kernighan traveling salesman heuristic. European Journal of Operational Research 2000;126; 106-130.

Herrera M, Torgo L, Izquierdo J, Pérez-García R. Predictive models for forecasting hourly urban water demand. Journal of Hydrology 2010;387; 141-150.

Higdon D, Kennedy M, Cavendish JC, Cafeo JA, Ryne RD. Combining field data and computer simulations for calibration and prediction. SIAM Journal on Scientific Computing 2004;26; 448-466.

Ho TK, 1995. Random decision forests, Document Analysis and Recognition, 1995., Proceedings Of The Third International IEEE Conference. IEEE, pp. 278-282.

Ho TK. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 1998;20; 832-844.

Hsu C-W, Chang C-C, Lin C-J, 2003. A practical guide to support vector classification, Department of Computer Science and Information Engineering, National Taiwan University.

Hu W, He Y, Liu Z, Tan J, Yang M, Chen J. Toward a digital twin: Time series prediction based on a hybrid ensemble empirical mode decomposition and BO-LSTM neural networks. Journal of Mechanical Design 2021;143; 051705.

Hu Z, Du X. Mixed efficient global optimization for time-dependent reliability analysis. Journal of Mechanical Design 2015;137; 051401.

Hu Z, Mahadevan S. A single-loop kriging surrogate modeling for time-dependent reliability analysis. Journal of Mechanical Design 2016;138; 061406.

Imrie C, Durucan S, Korre A. River flow prediction using artificial neural networks: Generalisation beyond the calibration range. Journal of Hydrology 2000;233; 138-153.

IPCC, Team CW. Climate change 2014: Synthesis report. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change 2014;27; 408-420.

Jack Williams RA, Janet K.Allen,Farrokh Mistree, 2020. Using Network Partitioning To Design a Green Supply Chain, IDETC, St.Louis, Mo,USA.

Jaeckel P, Rebonato R. The most general methodology for creating a valid correlation matrix for risk management and option pricing purposes. Journal of Risk 1999;2; 17-28.

Jafarian A, Asgari N, Mohri SS, Fatemi-Sadr E, Farahani RZ. The inventory-routing problem subject to vehicle failure. Transportation Research Part E: Logistics and Transportation Review 2019;126; 254-294.

Jagadeesh KA, Wenger AM, Berger MJ, Guturu H, Stenson PD, Cooper DN, Bernstein JA, Bejerano G. M-CAP eliminates a majority of variants of uncertain significance in clinical exomes at high sensitivity. Nature Genetics 2016;48; 1581.

Jägle E, 2007. Modelling of microstructural banding during transformations in steel, Department of Materials Science & Metallurgy. University of Cambridge, Cambridge, UK, pp. 1-63.

Jaillet P. A priori solution of a traveling salesman problem in which a random subset of the customers are visited. Operations Research 1988;36; 929-936.

Jain SKD, A.,, Srivastava DK. Application of ANN for reservoir inflow prediction and operation. Journal of Water Resources Planning and Management 1999;125; 263-271.

Jakhar SK. Performance evaluation and a flow allocation decision model for a sustainable supply chain of an apparel industry. Journal of Cleaner Production 2015;87; 391-413.

Jia L, Alizadeh R, Hao J, Wang G, Allen JK, Mistree F. A rule-based method for automated surrogate model selection. Advanced Engineering Informatics 2020;45; 101123.

Jiang C, Wei X, Huang Z, Liu J. An outcrossing rate model and its efficient calculation for time-dependent system reliability analysis. Journal of Mechanical Design 2017;139; 041402.

Jiang L, Chang H, Zhao S, Dong J, Lu W. A Travelling Salesman Problem With Carbon Emission Reduction in the Last Mile Delivery. IEEE Access 2019;7; 61620-61627.

Jiang Z, Chen S, Apley DW, Chen W. Reduction of epistemic model uncertainty in simulation-based multidisciplinary design. Journal of Mechanical Design 2016;138; 081403-081403-081413.

Jin M, Granda-Marulanda NA, Down I. The impact of carbon policies on supply chain design and logistics of a major retailer. Journal of Cleaner Production 2014;85; 453-461.

Jin R, Chen W, Simpson T, 2000. Comparative studies of metamodeling techniques under multiple modeling criteria, 8th Symposium on Multidisciplinary Analysis and Optimization. American Institute of Aeronautics and Astronautics; 2000.

Jin R, Chen W, Simpson TW. Comparative studies of metamodelling techniques under multiple modelling criteria. Structural and Multidisciplinary Optimization 2001;23; 1-13.

Jin Y. A comprehensive survey of fitness approximation in evolutionary computation. Soft Computing 2005;9; 3-12.

Jolliffe I. Principal component analysis: Wiley Online Library. Google Scholar 2005.

Jolliffe. I. Principal component analysis, 2nd edition ed. Springer: New York; 2002.

Jones DR. A taxonomy of global optimization methods based on response surfaces. Journal of Global Optimization 2001;21; 345-383.

Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. Journal of Global optimization 1998;13; 455-492.

Jones S, Bhadeshia H. Kinetics of the simultaneous decomposition of austenite into several transformation products. Acta Materialia 1997;45; 2911-2920.

Jones SJ, Bhadeshia HKDH, 2017. Program STRUCTURE on the Materials Algorithm Project web site. Cambridge University Press, United Kingdom.

Jung D, Kim JH. Water distribution system design to minimize costs and maximize topological and hydraulic reliability. Journal of Water Resources Planning and Management 2018;144; 06018005.

Kalayci CB, Kaya C. An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery. Expert Systems with Applications 2016;66; 163-175.

Kaleibari SS, Beiragh RG, Alizadeh R, Solimanpur M. A framework for performance evaluation of energy supply chain by a compatible network data envelopment analysis model. Scientia Iranica. Transaction E, Industrial Engineering 2016;23; 1904-1917.

Kannan D, Diabat A, Alrefaei M, Govindan K, Yong G. A carbon footprint based reverse logistics network design model. Resources, Conservation and Recycling 2012;67; 75-79.

Karimi-Googhari SH, Lee TS. Applicability of adaptive neuro-fuzzy inference systems in daily reservoir inflow forecasting. International Journal of Soft Computing 2011;6; 75-84.

Karson MJ, Manson AR, Hader RJ. Minimum bias estimation and experimental design for response surfaces. Technometrics 1969;11; 461-475.

Karwan MH, Rardin RL. Searchability of the composite and multiple surrogate dual functions. Operations Research 1980;28; 1251-1257.

Kathleen HVB, Cox DR. Some Systematic Supersaturated Designs. Technometrics 1962;4; 489-495.

Kaufman M, Balabanov V, Giunta AA, Grossman B, Mason WH, Burgee SL, Haftka RT, Watson LT. Variable-complexity response surface approximations for wing structural weight in HSCT design. Computational Mechanics 1996;18; 112-126.

Kennedy MC, O'Hagan A. Predicting the output from a complex computer code when fast approximations are available. Biometrika 2000;87; 1-13.

Kennedy MC, O'Hagan A. Bayesian calibration of computer models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2001;63; 425-464.

Keskin N, Plambeck EL. Greenhouse gas emissions accounting: allocating emissions from processes to co-products. Working Paper, Stanford Graduate School of Business 2011;201.

Khoshelham K, 2011. Accuracy analysis of kinect depth data, ISPRS workshop laser scanning, pp. 133-138.

Khuri A. A measure of rotatability for response-surface designs. Technometrics 1988;30; 95-104.

Khuri AI, Mukhopadhyay S. Response surface methodology. Wiley Interdisciplinary Reviews: Computational Statistics 2010;2; 128-149.

Kim S, Kim H. A new metric of absolute percentage error for intermittent demand forecasts. International Journal of Forecasting 2016;32; 669-679.

Kitjacharoenchai P, Min B-C, Lee S. Two echelon vehicle routing problem with drones in last mile delivery. International Journal of Production Economics 2019; 107598.

Kleijnen JP. Regression and Kriging metamodels with their experimental designs in simulation: a review. European Journal of Operational Research 2017;256; 1-16.

Kleinberg E. Stochastic discrimination. Annals of Mathematics and Artificial intelligence 1990;1; 207-239.

Kleinberg E. An overtraining-resistant stochastic modeling method for pattern recognition. The annals of statistics 1996;24; 2319-2349.

Kleinberg EM. On the algorithmic implementation of stochastic discrimination. IEEE Transactions on Pattern Analysis and Machine Intelligence 2000;22; 473-490.

Klose A, Drexl A. Facility location models for distribution system design. European Journal of Operational Research 2005;162; 4-29.

Koç Ç, Bektaş T, Jabali O, Laporte G. Thirty years of heterogeneous vehicle routing. European Journal of Operational Research 2016;249; 1-21.

Koch PN, Simpson TW, Allen JK, Mistree F. Statistical approximations for multidisciplinary design optimization: the problem of size. Journal of Aircraft 1999;36; 275-286.

Kök AG, Shang K, Yücel Ş. Impact of electricity pricing policies on renewable energy investments and carbon emissions. Management Science 2018;64; 131-148.

Korda AA, Mutoh Y, Miyashita Y, Sadasue T, Mannan S. In situ observation of fatigue crack retardation in banded ferrite–pearlite microstructure due to crack branching. Scripta Materialia 2006;54; 1835-1840.

Krauss C, Do XA, Huck N. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. European Journal of Operational Research 2017;259; 689-702.

Krauss GB. Solidification, segregation, and banding in carbon and alloy steels. Metallurgical and Materials Transactions A 2003;34; 781-792.

Kuo T-C, Tseng M-L, Chen H-M, Chen P-S, Chang P-C. Design and analysis of supply chain networks with low carbon emissions. Computational Economics 2018;52; 1353-1374.

Kusiak A, Larson N. Decomposition and representation methods in mechanical design. Journal of Mechanical Design 1995;117; 17-24.

Kuzmenko V, Uryasev S, 2019. Kantorovich–Rubinstein Distance Minimization: Application to Location Problems, Large Scale Optimization in Supply Chains and Smart Manufacturing. Springer; 2019. pp. 59-68.

La Fuente D, Andres R, 2016. Simulation Metamodeling with Gaussian Process: A Numerical Study, Industrial Engineering. North Carolina State University.

Laporte G. The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research 1992;59; 231-247.

Laporte G, Osman IH. Routing problems: A bibliography. Annals of Operations Research 1995;61; 227-262.

Laurent L, Le Riche R, Soulier B, Boucard P-A. An Overview of Gradient-Enhanced Metamodels with Applications. Archives of Computational Methods in Engineering 2017.

Lawler EL. The traveling salesman problem: a guided tour of combinatorial optimization. Wiley-Interscience Series in Discrete Mathematics 1985.

Lelièvre N, Beaurepaire P, Mattrand C, Gayton N. AK-MCSi: A Kriging-based method to deal with small failure probabilities and time-consuming models. Structural Safety 2018;73; 1-11.

Lemercier B, Lacoste M, Loum M, Walter C. Extrapolation at regional scale of local soil knowledge using boosted classification trees: A two-step approach. Geoderma 2012;171; 75-84.

LeSage J, Pace RK. Introduction to spatial econometrics. Chapman and Hall/CRC; 2009.

Levine N, Wachs M. Bus crime in Los Angeles: I—Measuring the incidence. Transportation Research Part A: General 1986;20; 273-284.

Lewis K, Mistree F. The other side of multidisciplinary design optimization: accommodating a multi-objective, uncertain and non-deterministic world. Engineering Optimization 1998;31; 161-189.

Li B, Yang G, Wan R, Dai X, Zhang Y. Comparison of random forests and other statistical methods for the prediction of lake water level: a case study of the Poyang Lake in China. Hydrology Research 2016a;47; 69-83.

Li E, Wang H, Ye F. Two-level multi-surrogate assisted optimization method for high dimensional nonlinear problems. Applied Soft Computing 2016b;46; 26-36.

Li G, Haining R, Richardson S, Best N. Space–time variability in burglary risk: a Bayesian spatio-temporal modelling approach. Spatial Statistics 2014;9; 180-191.

Li K, Deb K, Altinoz T, Yao X, 2017. Empirical investigations of reference point based methods when facing a massively large number of objectives: First results, International Conference on Evolutionary Multi-Criterion Optimization. Springer, pp. 390-405.

Li Y, Soleimani H, Zohal M. An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. Journal of Cleaner Production 2019;227; 1161-1172.

Liang C, Mahadevan S. Stochastic multidisciplinary analysis with high-dimensional coupling. AIAA Journal 2016;54; 1209-1219.

Liesenfeld R, Richard J-F, Vogler J, 2016. Likelihood evaluation of high-dimensional spatial latent Gaussian models with Non-Gaussian response variables, Spatial Econometrics: Qualitative and Limited Dependent Variables. Emerald Group Publishing Limited; 2016. pp. 35-77.

Lim D, Ong Y-S, Jin Y, Sendhoff B, 2007. A study on metamodeling techniques, ensembles, and multi-surrogates in evolutionary computation, Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp. 1288-1295.

Lin CD, Anderson-Cook CM, Hamada MS, Moore LM, Sitter RR. Using genetic algorithms to design experiments: a review. Quality and Reliability Engineering International 2015;31; 155-167.

Lin DK. A new class of supersaturated designs. Technometrics 1993;35; 28-31.

Liu B, Koziel S, Zhang Q. A multi-fidelity surrogate-model-assisted evolutionary algorithm for computationally expensive optimization problems. Journal of Computational Science 2016;12; 28-37.

Liu K, Tovar A, Nutwell E, Detwiler D, 2015. Thin-walled compliant mechanism component design assisted by machine learning and multiple surrogates, SAE 2015 World Congress & Exhibition. SAE International, Detroit, Michigan, pp. 1-16.

Lucas JM. Using response surface methodology to achieve a robust process. Journal of Quality Technology 1994;26; 248-260.

Lv Z, Zhao J, Wang W, Liu Q. A multiple surrogates based PSO algorithm. Artificial Intelligence Review 2018;5; 45-68.

Lv Z, Zhao J, Wang W, Liu Q. A multiple surrogates based PSO algorithm. Artificial Intelligence Review 2019;52; 2169-2190.

MacCalman A, Lesinski G, Goerger S. Integrating External Simulations Within the Model-Based Systems Engineering Approach Using Statistical Metamodels. Procedia Computer Science 2016;95; 436-441.

Mack Y, Goel T, Shyy W, Haftka R, Queipo N, 2005. Multiple surrogates for the shape optimization of bluff body-facilitated mixing, 43rd AIAA Aerospace Sciences Meeting and Exhibit. AIAA, Reno, Nevada, pp. 1-22.

Madala HR, Ivakhnenko AG. Inductive learning algorithms for complex systems modeling. CRC Press Boca Raton: NY; 1994.

Maier HR, Dandy GC. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environmental modelling & software 2000;15; 101-124.

Makridakis S, Wheelwright S, Hyndman R, Chang Y. Forecasting methods and applications, 3rd ed. ed. John Wiley & Sons: New York; 1998.

McKay MD, Beckman RJ, Conover WJ. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 1979;21; 239-245.

Mehmani A, Chowdhury S, Meinrenken C, Messac A. Concurrent surrogate model selection (COSMOS): optimizing model type, kernel function, and hyper-parameters. Structural & Multidisciplinary Optimization 2018;57; 1093-1114.

Mentzer JT, DeWitt W, Keebler JS, Min S, Nix NW, Smith CD, Zacharia ZG. Defining supply chain managment Journal of Business Logistics 2001;22; 1-25.

Messer M, Panchal JH, Krishnamurthy V, Klein B, Yoder PD, Allen JK, Mistree F. Model selection under limited information using a value-of-information-based indicator. Journal of Mechanical Design 2010;132; 121008.

Michalski RS, 1983. A theory and methodology of inductive learning, Machine Learning, Volume I. Elsevier; 1983. pp. 83-134.

Mirjalili S, 2019. Evolutionary Radial Basis Function Networks, In: Mirjalili S (Ed), Evolutionary Algorithms and Neural Networks: Theory and Applications, vol. 2. Springer International Publishing, Cham; 2019. pp. 105-139.

Mirzapour Al-e-hashem SM, Rekik Y, Hoseinhajlou EM. A hybrid L-shaped method to solve a bi-objective stochastic transshipment-enabled inventory routing problem. International Journal of Production Economics 2017;30; 1-18.

Mistree F, Hughes O, Brass B, 1992. The Compromise Decision Support Problem and the Adaptive Linear Programing Algorithem, Structural Optimization: Status and Promise. AIAA, Washington D.C; 1992. pp. 247-286.

Mohajeri A, Fallah M. A carbon footprint-based closed-loop supply chain model under uncertainty with risk analysis: A case study. Transportation Research Part D: Transport and Environment 2016;48; 425-450.

Mohammadi K, Eslami HR, Dardashti SD. Comparison of regression, ARIMA and ANN models for reservoir inflow forecasting using snowmelt equivalent (a case study of Karaj). Journal of Agricultural Science and Technology 2005;7; 17-30.

Mohebalizadehgashti F, Zolfagharnia H, Amin SH. Designing a green meat supply chain network: A multi-objective approach. International Journal of Production Economics 2020;219; 312-327.

Montgomery DC. Design and analysis of experiments. John Wiley & Sons; 2017.

Mugunthan P, Shoemaker CA. Assessing the impacts of parameter uncertainty for computationally expensive groundwater models. Water Resources Research 2006;42.

Muluye GY, Coulibaly P. Seasonal reservoir inflow forecasting with low-frequency climatic indices: a comparison of data-driven methods. Hydrological Sciences Journal 2007;52; 508-522.

Myers RH, Montgomery DC, Anderson-Cook CM. Response surface methodology: process and product optimization using designed experiments. John Wiley & Sons; 2016.

Nakamura K, Tita G, Krackhardt D. Violence in the “balance”: a structural analysis of how rivals, allies, and third-parties shape inter-gang violence. Global Crime 2019; 1-25.

Narayan PK, Smyth* R. Crime rates, male youth unemployment and real income in Australia: evidence from Granger causality tests. Applied Economics 2004;36; 2079-2095.

Nellippallil AB, Rangaraj V, Gautham BP, Singh AK, Allen JK, Mistree F. An inverse, decision-based design method for integrated design exploration of materials, products and manufacturing processes. ASME Journal of Mechanical Design 2018;140; 111403-111417.

Neves-Moreira F, Almada-Lobo B, Cordeau J-F, Guimarães L, Jans R. Solving a large multi-product production-routing problem with delivery time windows. Omega 2019;86; 154-172.

Nguyen N-K. An algorithmic approach to constructing supersaturated designs. Technometrics 1996;38; 69-73.

Niu Y, Yang Z, Chen P, Xiao J. Optimizing the green open vehicle routing problem with time windows by minimizing comprehensive routing cost. Journal of Cleaner Production 2018;171; 962-971.

Nouira I, Hammami R, Frein Y, Temponi C. Design of forward supply chains: Impact of a carbon emissions-sensitive demand. International Journal of Production Economics 2016;173; 80-98.

Osmani A, Zhang J. Economic and environmental optimization of a large scale sustainable dual feedstock lignocellulosic-based bioethanol supply chain in a stochastic environment. Applied Energy 2014;114; 572-587.

Othman F, Naseri M. Reservoir inflow forecasting using artificial neural network. International Journal Of Physical Sciences 2011;6; 434-440.

Owen AB. Orthogonal arrays for computer experiments, integration and visualization. Statistica Sinica 1992; 439-452.

Padberg M, Rinaldi G. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. SIAM Review 1991;33; 60-100.

Pahl G, W. Beitz, J. Feldhusen, and K.H. Grote. Engineering design, A Systematic Approach. Springer; 2007.

Paiva RM, D. Carvalho AR, Crawford C, Suleman A. Comparison of surrogate models in a multidisciplinary optimization framework for wing design. AIAA Journal 2010;48; 995-1006.

Panchal JH, Kalidindi SR, McDowell DL. Key computational modeling issues in integrated computational materials engineering. Computer-Aided Design 2013;45; 4-25.

Panchal JH, Paredis CJ, Allen JK, Mistree F. Managing design-process complexity: a value-of-information based approach for scale and decision decoupling. Journal of Computing and Information Science in Engineering 2009;9; 021005.

Panchal JH, Paredis CJJ, Allen JK, Mistree F. A value-of-information based approach to simulation model refinement. Engineering Optimization 2008;40; 223-251.

Pancras J, Sriram S, Kumar V. Empirical investigation of retail expansion and cannibalization in a dynamic environment. Management Science 2012;58; 2001-2018.

Papachristos AV, Bastomski S. Connected in crime: The enduring effect of neighborhood networks on the spatial patterning of violence. American Journal of Sociology 2018;124; 517-568.

Passos A, Luersen M. Multiobjective optimization of laminated composite parts with curvilinear fibers using Kriging-based approaches. Structural and Multidisciplinary Optimization 2018;57; 1115-1127.

Pedersen K, Emblemsvag J, Bailey R, Allen J, Mistree F, 2000. The'Validation Square'-Validating Design Methods, ASME Design Theory and Methodology Conference. ASME, NY.

Pektas AO, Cigizoglu HK. Ann hybrid model versus ARIMA and ARIMAX models of runoff coefficient. Journal of Hydrology 2013;500; 21-36.

Peng Q, Wang C, Xu L. Emission abatement and procurement strategies in a low-carbon supply chain with option contracts under stochastic demand. Computers & Industrial Engineering 2020;144; 106502.

Persad RA. Hierarchical Bayesian modeling for the spatial analysis of robberies in Toronto, Canada. Spatial Information Research 2019; 1-13.

Petersen N, Ward G. The transmission of historical racial violence: Lynching, civil rights–era terror, and contemporary interracial homicide. Race and Justice 2015;5; 114-143.

Peterson RD, Krivo LJ, Browning CR, 2017. Segregation and race/ethnic inequality in crime: New directions, Taking Stock. Routledge; 2017. pp. 169-187.

Phoa FKH, Pan Y-H, Xu H. Analysis of supersaturated designs via the Dantzig selector. Journal of Statistical Planning and Inference 2009;139; 2362-2372.

Picheny V, Ginsbourger D, Roustant O, Haftka RT, Kim N-H. Adaptive designs of experiments for accurate approximation of a target region. Journal of Mechanical Design 2010;132; 071008-071008-071009.

Pineda-Ríos W, Giraldo R, Porcu E. Functional SAR models: With application to spatial econometrics. Spatial Statistics 2019;29; 145-159.

Pishvaee MS, Torabi SA, Razmi J. Credibility-based fuzzy mathematical programming model for green logistics design under uncertainty. Computers & Industrial Engineering 2012;62; 624-632.

Plambeck EL. Reducing greenhouse gas emissions through operations and supply chain management. Energy Economics 2012;34; S64-S74.

Potvin J-Y. Genetic algorithms for the traveling salesman problem. Annals of Operations Research 1996;63; 337-370.

Pramanik N, Panda RK. Application of neural network and adaptive neuro-fuzzy inference systems for river flow prediction. Hydrological Sciences Journal 2009;54; 247-260.

Probst JC, Glover S, Kirksey V. Strange Harvest: a Cross-sectional Ecological Analysis of the Association Between Historic Lynching Events and 2010–2014 County Mortality Rates. Journal of Racial and Ethnic Health Disparities 2019;6; 143-152.

Purohit AK, Shankar R, Dey PK, Choudhary A. Non-stationary stochastic inventory lot-sizing with emission and service level constraints in a carbon cap-and-trade system. Journal of Cleaner Production 2016;113; 654-661.

Qi Q, Wang J, Bai Q. Pricing decision of a two-echelon supply chain with one supplier and two retailers under a carbon cap regulation. Journal of Cleaner Production 2017;151; 286-302.

Qian PZ, Tang B, Wu CJ. Nested space-filling designs for computer experiments with two levels of accuracy. Statistica Sinica 2009; 287-300.

Qian PZG, Wu H, Wu CJ. Gaussian process models for computer experiments with qualitative and quantitative factors. Technometrics 2008;50; 383-396.

Qian Z, Seepersad CC, Joseph VR, Allen JK, Wu CJ. Building surrogate models based on detailed and approximate simulations. Journal of Mechanical Design 2006;128; 668-677.

Qiu Q, Li B, Feng P. Optimal design of hydraulic excavator working device based on multiple surrogate models. Advances in Mechanical Engineering 2016;8; 1687814016647947.

R. CJ. AI in CAI: an artificial-intelligence approach to computer-assisted instruction. IEEE Transactions on Man-Machine Systems 1970;11; 190-202.

Rand, 2009. Neighborhood effects on crime and youth violence : the role of business improvement districts in Los Angeles, Rand Infrastructure, Safety Environment, Centers for Disease, Control Prevention, Rand, Corporation, NY, pp. 1-256.

Ratcliffe J, 2010. Crime mapping: Spatial and temporal challenges, Handbook of Quantitative Criminology. Springer, Cleveland, Ohio; 2010. pp. 5-24.

Ratto M, Castelletti A, Pagano A, 2012. Emulation techniques for the reduction and sensitivity analysis of complex environmental models. Elsevier.

Razavi S, Tolson BA, Burn DH. Review of surrogate modeling in water resources. Water Resources Research 2012;48; 1-32.

Reese CS, Wilson AG, Hamada M, Martz HF, Ryan KJ. Integrated analysis of computer and physical experiments. Technometrics 2004;46; 153-164.

Regis RG, Shoemaker CA. A stochastic radial basis function method for the global optimization of expensive functions. INFORMS Journal on Computing 2007;19; 497-509.

Rehman Khan SA, 2018. Introductory chapter: introduction of green supply chain management, Green Practices and Strategies in Supply Chain Management. IntechOpen; 2018.

Reyes D, Savelsbergh M, Toriello A. Vehicle routing with roaming delivery locations. Transportation Research Part C: Emerging Technologies 2017;80; 71-91.

Ridgeway G, MacDonald JM. Effect of rail transit on crime: a study of Los Angeles from 1988 to 2014. Journal of Quantitative Criminology 2017;33; 277-291.

Ripley BD. Pattern recognition and neural networks. Cambridge university press; 2007.

Roth RE, Ross KS, Finch BG, Luo W, MacEachren AM. Spatiotemporal crime analysis in US law enforcement agencies: Current practices and unmet needs. Government Information Quarterly 2013;30; 226-240.

Ruiz E, Soto-Mendoza V, Barbosa AER, Reyes R. Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm. Computers & Industrial Engineering 2019;133; 207-219.

Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. Statistical Science 1989;4; 409-435.

Sadaghiani M, Alizadeh R, Bahrami M, 2014. Scenario-based planning for energy foresight case study: Iran's transportation industry, The 10th international Energy Conference (IEC 2014), pp. 1-26.

Salavati-Khoshghalb M, Gendreau M, Jabali O, Rei W. An exact algorithm to solve the vehicle routing problem with stochastic demands under an optimal restocking policy. European Journal of Operational Research 2019;273; 175-189.

Samad A, Kim K-Y, Goel T, Haftka RT, Shyy W, 2006. Shape optimization of turbomachinery blade using multiple surrogate models, ASME 2006 2nd Joint U.S.-European Fluids Engineering Summer Meeting Collocated With the 14th International Conference on Nuclear Engineering. ASME, Miami, Florida, USA, pp. 827-836.

Samad A, Lee K-D, Kim K-Y, Haftka R, 2007. Application of multiple-surrogate model to optimization of a dimpled channel, 7th World Congresses of Structural and Multidisciplinary Optimization. WCSMO, Seoul, Korea, pp. 2276-2282.

Sargent RG, 2010. Verification and validation of simulation models, Proceedings of the 2010 Winter Simulation Conference. IEEE, pp. 166-183.

Schmidhuber J. Deep learning in neural networks: An overview. Neural Networks 2015;61; 85-117.

Scholz-Reiter B, Frazzon EM, Makuschewitz T. Integrating manufacturing and logistic systems along global supply chains. CIRP Journal of Manufacturing Science and Technology 2010a;2; 216-223.

Scholz-Reiter B, Schwindt C, Makuschewitz T, Frazzon EM, 2010b. An approach for the integration of production scheduling and inter-facility transportation within global supply chains, Proceedings of POMS Annual Conference, Vancouver, CA.

Sechi GM, Gaivoronski AA, Napolitano J. Optimising pumping activation in multi-reservoir water supply systems under uncertainty with stochastic quasi-gradient methods. Water Resources Management 2019;33; 1881-1895.

Seepersad CC, Pedersen K, Emblemsvåg J, Bailey R, Allen JK, Mistree F. The validation square: how does one verify and validate a design method. Decision Making in Engineering Design 2006; 303-314.

Seni G, Elder JF. Ensemble methods in data mining: improving accuracy through combining predictions. Synthesis Lectures on Data Mining and Knowledge Discovery 2010;2; 1-126.

Seuring S, Müller M. From a literature review to a conceptual framework for sustainable supply chain management. Journal of Cleaner Production 2008;16; 1699-1710.

Shafiei Kaleibari S, Gharizadeh Beiragh R, Alizadeh R, Solimanpur M. A framework for performance evaluation of energy supply chain by a compatible network data envelopment analysis model. Scientia Iranica 2016;23; 1904-1917.

Shalamu A, 2009. Monthly and seasonal streamflow forecasting in the Rio Grande Basin, Civil Engineering. New Mexico State University, New Mexico, USA, p. 321.

Shan S, Wang GG. Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. Structural and Multidisciplinary Optimization 2010;41; 219-241.

Shankar Bhattacharjee K, Kumar Singh H, Ray T. Multi-objective optimization with multiple spatially distributed surrogates. Journal of Mechanical Design 2016;138; 091401-091401-091410.

Shaw K, Irfan M, Shankar R, Yadav SS. Low carbon chance constrained supply chain network design problem: a Benders decomposition based approach. Computers & Industrial Engineering 2016;98; 483-497.

Shi R, Liu L, Long T, Liu J. An efficient ensemble of radial basis functions method based on quadratic programming. Engineering Optimization 2016;48; 1202-1225.

Shi Y, Lu Z, Zhang K, Wei Y. Reliability analysis for structures with multiple temporal and spatial parameters based on the effective first-crossing point. Journal of Mechanical Design 2017;139; 121403.

Shoup D. The high cost of free parking: Updated edition. Taylor & Francis: New York, USA; 2017.

Shukla S, Sheffield J, Wood EF, Lettenmaier DP. On the sources of global land surface hydrologic predictability. Hydrology and Earth System Sciences 2013;17.

Shyy W, Tucker PK, Vaidyanathan R. Response surface and neural network techniques for rocket engine injector optimization. Journal of Propulsion and Power 2001;17; 391-401.

Sim DYY, Teh CS, Ismail AI. Improved boosted decision tree algorithms by adaptive apriori and post-pruning for predicting obstructive sleep apnea. Advanced Science Letters 2018;24; 1680-1684.

Simpson T, Mistree F, Korte J, Mauery T, 1998. Comparison of response surface and kriging models for multidisciplinary design optimization, 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. American Institute of Aeronautics and Astronautics; 1998.

Simpson T, Peplinski J, Koch PN, Allen JK. Metamodels for computer-based engineering design: survey and recommendations. Engineering with Computers 2001;17; 129-150.

Simpson T, Toropov V, Balabanov V, Viana F, 2008. Design and analysis of computer experiments in multidisciplinary design optimization: A review of how far we have come-or not, 12th AIAA/ISSMO multidisciplinary analysis and optimization conference, p. 5802.

Simpson TW, Booker AJ, Ghosh D, Giunta AA, Koch PN, Yang R-J. Approximation methods in multidisciplinary analysis and optimization: a panel discussion. Structural and Multidisciplinary Optimization 2004;27; 302-313.

Simpson TW, Poplinski JD, Koch PN, Allen JK, 1997. On the use of statistics in design and the implications for deterministic computer experiments, ASME Design Engineering Technical Conferences, Sacramento, California, pp. 14-17.

Singaravel S, Suykens J, Geyer P. Deep convolutional learning for general early design stage prediction models. Advanced Engineering Informatics 2019;42; 100982.

Smith WF, Milisavljevic J, Sabeghi M, Allen JK, Mistree F, 2015. The realization of engineered systems with considerations of complexity, International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, p. V007T006A019.

Snyder LV, Shen Z-JM. Fundamentals of supply chain theory. John Wiley: Hoboken, New Jersey; 2011.

Sobieszczanski-Sobieski J, Haftka RT. Multidisciplinary aerospace design optimization: survey of recent developments. Structural Optimization 1997;14; 1-23.

Soleimani H, Chaharlang Y, Ghaderi H. Collection and distribution of returned-remanufactured products in a vehicle routing problem with pickup and delivery considering sustainable and green criteria. Journal of Cleaner Production 2018;172; 960-970.

Soltanisehat L, Alizadeh R, Hao H, Choo K-KR. Technical, temporal, and spatial research challenges and opportunities in blockchain-based healthcare: a systematic literature review. IEEE Transactions on Engineering Management 2020;1; 1-25.

Soltanisehat L, Alizadeh R, Mehregan N. Research and Development Investment and Productivity Growth in Firms with Different Levels of Technology. Iranian Economic Review 2018;23.

Song X, Lv L, Li J, Sun W, Zhang J. An advanced and robust ensemble surrogate model: extended adaptive hybrid functions. Journal of Mechanical Design 2018;140; 041402-041409.

Soysal M, Bloemhof-Ruwaard JM, van der Vorst JGAJ. Modelling food logistics networks with emission considerations: The case of an international beef supply chain. International Journal of Production Economics 2014;152; 57-70.

Soysal M, Çimen M, Belbağ S, Toğrul E. A review on sustainable inventory routing. Computers & Industrial Engineering 2019.

Spitzig W. Effect of sulfide inclusion morphology and pearlite banding on anisotropy of mechanical properties in normalized C-Mn steels. Metallurgical Transactions A 1983;14; 271-283.

Stein ML. Interpolation of spatial data: some theory for kriging. Springer Science & Business Media; 2012.

Talebizadeh M, Moridnejad A. Uncertainty analysis for the forecast of lake level fluctuations using ensembles of ann and ANFIS models. Expert Systems with Applications 2011;38; 4126-4135.

Tao F, Cheng J, Qi Q, Zhang M, Zhang H, Sui F. Digital twin-driven product design, manufacturing and service with big data. The International Journal of Advanced Manufacturing Technology 2018;94; 3563-3576.

Tao ZG, Guang ZY, Hao S, Song HJ, Xin DG. Multi-period closed-loop supply chain network equilibrium with carbon emission constraints. Resources, Conservation and Recycling 2015;104; 354-365.

Tita GE, Greenbaum RT, 2009. Crime, neighborhoods, and units of analysis: putting space in its place, Putting Crime in its Place. Springer; 2009. pp. 145-170.

Tita GE, Radil SM, 2010. Spatial regression models in criminology: Modeling social processes in the spatial weights matrix, Handbook of Quantitative Criminology. Springer; 2010. pp. 101-121.

Toal DJ, Bressloff N, Keane A, Holden C. The development of a hybridized particle swarm for kriging hyperparameter tuning. Engineering Optimization 2011;43; 675-699.

Tolnay SE, Deane G, Beck EM. Vicarious violence: Spatial effects on southern lynchings, 1890-1919. American Journal of Sociology 1996;102; 788-815.

Tomita Y. Effect of modified austemper on tensile properties of 0· 52% C steel. Materials Science and Technology 1995;11; 994-997.

Torabi SHR, Alibabaei S, Bonab BB, Sadeghi MH, Faraji G. Design and optimization of turbine blade preform forging using RSM and NSGA II. Journal of Intelligent Manufacturing 2017;28; 1409-1419.

Trucano TG, Swiler LP, Igusa T, Oberkampf WL, Pilch M. Calibration, validation, and sensitivity analysis: What's what. Reliability Engineering & System Safety 2006;91; 1331-1357.

Tuck M, Riley D, 2017. The theory of reasoned action: A decision theory of crime, The Reasoning Criminal. Routledge; 2017. pp. 156-169.

Tucker CS, Kim HM. Trend mining for predictive product design. Journal of Mechanical Design 2011;133; 111008.

Tutum CC, Deb K, 2015. A multimodal approach for evolutionary multi-objective optimization (MEMO): proof-of-principle results, International Conference on Evolutionary Multi-Criterion Optimization. Springer, pp. 3-18.

Uchoa E, Pecin D, Pessoa A, Poggi M, Vidal T, Subramanian A. New benchmark instances for the capacitated vehicle routing problem. European Journal of Operational Research 2017;257; 845-858.

Valipour M, Banihabib M, Behbahani S. Monthly inflow forecasting using autoregressive artificial neural network. Journal of Applied Sciences 2012;12; 2139-2147.

Vapnik V. The nature of statistical learning theory. Springer science & business media; 2013.

Varadarajan S, Chen∗ WEI, Pelka CJ. Robust Concept Exploration of Propulsion Systems with Enhanced Model Approximation Capabilities. Engineering Optimization 2000;32; 309-334.

Venter G, Haftka R, Chirehdast M, Venter G, Haftka R, Chirehdast M, 1997. Response surface approximations for fatigue life prediction, 38th Structures, Structural Dynamics, and Materials Conference. American Institute of Aeronautics and Astronautics; 1997.

Viana F, Haftka R, Steffen JV, Butkewitsch S, F Leal M, 2018. Optimal use of multiple surrogate for reduced RMS error in meta-model, NSF Engineering Research and Innovation Conference, Knoxville, Tennessee, pp. 1-13.

Viana F, Haftka R, Watson L, 2010a. Why not run the efficient global optimization algorithm with multiple surrogates?, 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 18th AIAA/ASME/AHS Adaptive Structures Conference 12th, p. 3090.

Viana FA, Haftka RT, Watson LT. Efficient global optimization algorithm assisted by multiple surrogate techniques. Journal of Global Optimization 2013;56; 669-689.

Viana FA, Picheny V, Haftka RT, 2009. Conservative prediction via safety margin: design through cross-validation and benefits of multiple surrogates, ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, pp. 741-750.

Viana FA, Picheny V, Haftka RT. Using cross validation to design conservative surrogates. AIAA Journal 2010b;48; 2286-2298.

Viana FAC, and R. T. Haftka, 2008. Using multiple surrogates for metamodeling, Proceedings of the 7th ASMO-UK/ISSMO International Conference on Engineering Design Optimization, edited by V. V. Toropov et al., Int. Soc. for Struct. And Multidiscip. Optim ed, pp. 1–18.

Viana FAC, Haftka RT, 2008. Using multiple surrogates for minimization of the RMS Error in meta-modeling, ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. ASME, Brooklyn, New York, USA, pp. 851-860.

Villanueva D, Haftka RT, Le Riche R, Picard G, 2013. Locating multiple candidate designs with surrogate-based optimization, 10th World Congress on Structural and Multidisciplinary Optimization, Orlando, USA, May. Citeseer, pp. 20-24.

Vincent FY, Redi AP, Hidayat YA, Wibowo OJ. A simulated annealing heuristic for the hybrid vehicle routing problem. Applied Soft Computing 2017;53; 119-132.

Vinzi V, Chin WW, Henseler J, Wang H, 2010. Handbook of partial least squares. Springer.

W. Bandler J, Koziel S, Madsen K. Editorial - Surrogate modeling and space mapping for engineering optimization; 2008.

Walmart, 2018a. 2018 Global Responsibility Report, Arkansas, USA.

Walmart, 2018b. 2018 Global Responsibility Report, Arkansas, USA, p. 127.

Waltho C, Eldedhli S, Gzara F. Green supply chain network design: A review focused on policy adoption and emission quantification. International Journal of Production Economics 2019;208; 305-318.

Wang B, Zhang D, Zhang D, Brantingham PJ, Bertozzi AL. Deep learning for real time crime forecasting. arXiv preprint arXiv:1707.03340 2017.

Wang C, Duan Q, Gong W, Ye A, Di Z, Miao C. An evaluation of adaptive surrogate modeling based optimization with two benchmark problems. Environmental Modelling & Software 2014;60; 167-179.

Wang GG, Shan S. Review of metamodeling techniques in support of engineering design optimization. Journal of Mechanical Design 2006;129; 370-380.

Wang H. Routing and scheduling for a last-mile transportation system. Transportation Science 2019;53; 131-147.

Wang H, Ye F, Li E, Li G. A comparative study of expected improvement-assisted global optimization with different surrogates. Engineering Optimization 2016;48; 1432-1458.

Wang R, Nellippallil AB, Wang G, Yan Y, Allen JK, Mistree F. Systematic design space exploration using a template-based ontological method. Advanced Engineering Informatics 2018;36; 163-177.

Wang Z, Wang P. A nested extreme response surface approach for time-dependent reliability-based design optimization. Journal of Mechanical Design 2012;134; 121007.

Wei L, Zhang Z, Zhang D, Leung SC. A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research 2018;265; 843-859.

Wieland A, Wallenburg CM. Dealing with supply chain risks: Linking risk management practices and strategies to performance. International Journal of Physical Distribution & Logistics Management 2012;42; 887-905.

Wikiland, 2020. Municipalities of Puerto rico. Wikiland, United Kingdom.

Williams BA, Cremaschi S. Surrogate model selection for design space approximation and surrogatebased optimization. Computer Aided Chemical Engineering 2019;47; 353-358.

Williams J, Alizadeh R, Allen JK, Mistree F, 2020. Using network partitioning to design a green supply chain, International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, p. V11BT11A050.

Wilson JQ, Kelling GL. Broken windows: the police and neighborhood safety. Criminological Perspectives: Essential Readings 2003;400.

Witten IH, Frank E, Hall MA, Pal CJ. Data mining: practical machine learning tools and techniques. Morgan Kaufmann: Burlington, MA; 2016.

Wo JC. Mixed land use and neighborhood crime. Social Science Research 2019;78; 170-186.

Woolsey RC. Crime and Punishment: Los Angeles County, 1850-1856. Southern California Quarterly 1979;61; 79-98.

Worland SC, Farmer WH, Kiang JE. Improving predictions of hydrological low-flow indices in ungaged basins using machine learning. Environmental Modelling & Software 2018;101; 169-182.

Wright EM, Skubak Tillyer M. Neighborhoods and intimate partner violence against women: the direct and interactive effects of social ties and collective efficacy. Journal of Interpersonal Violence 2017; 0886260517712276.

Wu C, Chau K, Li Y. Methods to improve neural network performance in daily flows prediction. Journal of Hydrology 2009;372; 80-93.

Xavier VL, Xavier AE. Accelerated hyperbolic smoothing method for solving the multisource Fermat–Weber and k-Median problems. Knowledge-Based Systems 2019; 105226.

Xiao Y, Konak A. A genetic algorithm with exact dynamic programming for the green vehicle routing & scheduling problem. Journal of Cleaner Production 2017;167; 1450-1463.

Xiong S, Qian PZ, Wu CJ. Sequential design and analysis of high-accuracy and low-accuracy computer codes. Technometrics 2013;55; 37-46.

Xiong Y, Chen W, Tsui K-L, Apley DW. A better understanding of model updating strategies in validating engineering models. Computer Methods in Applied Mechanics and Engineering 2009;198; 1327-1337.

Xu J, Zeger SL. The evaluation of multiple surrogate endpoints. Biometrics 2001;57; 81-87.

Xue Z, Lietz R, Rigoni E, Parashar S, Kansara S, 2013. RSM improvement methods for computationally expensive industrial cae analysis, 10th World Congress on Structural and Multidisciplinary Optimization, Orlando, Florida, USA.

Yang RJ, Gu L, Liaw L, Gearhart C, Tho CH, Liu X, Wang BP, 2000. Approximations for safety optimization of large systems, ASME Design Automation Conference, Baltimore, MD.

Yang T, Lin H-C, Chen M-L. Metamodeling approach in solving the machine parameters optimization problem using neural network and genetic algorithms: A case study. Robotics and Computer-Integrated Manufacturing 2006;22; 322-331.

Yin H, Fang H, Wen G, Gutowski M, Xiao Y. On the ensemble of metamodels with multiple regional optimized weight factors. Structural and Multidisciplinary Optimization 2018;5; 22-49.

Yu H, Tan Y, Sun C, Zeng J. A comparison of quality measures for model selection in surrogate-assisted evolutionary algorithm. Soft Computing 2019;23; 12417-12436.

Yu S, Wang Z. A novel time-variant reliability analysis method based on failure processes decomposition for dynamic uncertain structures. Journal of Mechanical Design 2018;140; 051401.

Zakeri A, Dehghanian F, Fahimnia B, Sarkis J. Carbon pricing versus emissions trading: A supply chain planning perspective. International Journal of Production Economics 2015;164; 197-205.

Zaletelj V, Vrabič R, Hozdić E, Butala P. A foundational ontology for the modelling of manufacturing systems. Advanced Engineering Informatics 2018;38; 129-141.

Zamani Sabzi H, Abudu S, Alizadeh R, Soltanisehat L, Dilekli N, King JP. Integration of time series forecasting in a dynamic decision support system for multiple reservoir management to conserve water sources. Energy Sources, Part A: Recovery, Utilization, and Environmental Effects 2018;40; 1398-1416.

Zerpa LE, Queipo NV, Pintos S, Salager J-L. An optimization methodology of alkaline–surfactant–polymer flooding processes using field scale numerical simulation and multiple surrogates. Journal of Petroleum Science and Engineering 2005;47; 197-208.

Zhang G, Allaire D, McAdams DA, Shankar V. Generating technology evolution prediction intervals using a bootstrap method. Journal of Mechanical Design 2019;141; 061401.

Zhou L, Baldacci R, Vigo D, Wang X. A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution. European Journal of Operational Research 2018a;265; 765-778.

Zhou Q, Jiang P, Shao X, Hu J, Cao L, Wan L. A variable fidelity information fusion method based on radial basis function. Advanced Engineering Informatics 2017;32; 26-39.

Zhou Q, Qian PZG, Zhou S. A simple approach to emulation for computer models with qualitative and quantitative factors. Technometrics 2011;53; 266-273.

Zhou Q, Shao X, Jiang P, Gao Z, Wang C, Shu L. An active learning metamodeling approach by sequentially exploiting difference information from variable-fidelity models. Advanced Engineering Informatics 2016;30; 283-297.

Zhou Q, Wang Y, Choi S, Jiang P, Shao X, Hu J, Shu L. A robust optimization approach based on multi-fidelity metamodel. Structural and Multidisciplinary Optimization 2018b;57; 775-797.

Zhou Y, Hu F, Zhou Z. Pricing decisions and social welfare in a supply chain with multiple competing retailers and carbon tax policy. Journal of Cleaner Production 2018c;190; 752-777.

Zhu J, Boyaci T, Ray S. Effects of upstream and downstream mergers on supply chain profitability. European Journal of Operational Research 2016;249; 131-143.