UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

A BI-OBJECTIVE FORMULATION FOR ROBUST DEFENSE STRATEGIES IN MULTI-COMMODITY NETWORKS: APPLICATION TO RAIL FREIGHT

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

MATTHEW MCCARTER
Norman, Oklahoma
2017

A BI-OBJECTIVE FORMULATION FOR ROBUST DEFENSE STRATEGIES IN MULTI-COMMODITY NETWORKS: APPLICATION TO RAIL FREIGHT

A THESIS APPROVED FOR THE
SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY

_____
Dr. Kash Barker, Chair

_____
Dr. Charles Nicholson

_____
Dr. Shivakumar Raman

To Bob, who started me on this journey;

To my parents, to whom I owe everything;

And to Conner, in whom I see all the things I love in this world.

Acknowledgements

I would first like to thank my advisor, Dr. Kash Barker, who, in his own way, has built me up and encouraged me to think critically and carefully (even about American composers).

Gratitude is also due to my colleague Mackenzie Whitman for her help in understanding the data involved in this work, and to Dr. Janet Allen, for her help in framing this study.

I would also like to thank the Industrial and Systems Engineering faculty. The dedicated and committed professors and administrators have crafted a well-rounded program and have been nothing but encouraging and supportive of myself and my colleagues. I feel like a part of a wonderful, quirky family, and there are few words I can give to describe what that means to me, especially given my former experience with academia.

Finally, I would like to thank my family for their patience, my fiancé for his support and sanity, my mentor Bob for his perspectives and guidance, and Dr. Charles Nicholson for what was probably the most salient advice I have yet received:

“Any problem worth solving is worth thinking about.”

Table of Contents

List of Tables

List of Figures

Abstract

Characterizing system performance under disruption is a growing area of research, particularly for describing a system's resilience to a disruption event. Within the framework of system resilience, this study approaches the minimization of a multiple-commodity system's vulnerability to multiple disruption events. The vulnerability of a system is defined by the degree to which commodities can no longer flow through the system to satisfy demand given a disruptive event. A multi-objective formulation is developed to find defense strategies at minimal cost that maintain a high level of demand satisfaction across all commodities. A solution method involving an estimation of the Pareto frontier via the Non-dominated Sorted Genetic Algorithm II (NSGA-II) is also proposed. A decision support environment is proposed and supported by application of the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS). The proposed formulation and solution method are illustrated with an example generated from the multi-commodity Swedish rail network.

Keywords: vulnerability, resilience, multi-commodity network flow, max flow, rail transportation, multiple commodity

Chapter 1: Introduction and Motivation

Characterizing the performance of critical infrastructure following a disruptive event is an increasingly important area of research, given (i) the frequency of possible disruptions, and (ii) the scale of their implications. The US government emphasizes resilience planning for critical infrastructures, suggesting that they "must be secure and able to withstand and rapidly recover from all hazards"[1]. The ability to withstand, to adapt to, and to recover from a disruption is generally referred to as resilience[2].

A number of qualitative and quantitative approaches for characterizing resilience have been offered in the recent literature[3]. One such approach is depicted graphically in Figure 1[4]-[6]. This approach describes system performance before, during, and after a disruption with function $\varphi(t)$. Note two dimensions of resilience in Figure 1. The lack of ability of the network to maintain performance immediately following disruptive event e^k is referred to as its *vulnerability*, an area receiving attention in the network literature for several years[7],[8]. The ability of the network to return to an acceptable level of performance in a timely manner is referred to as its *recoverability*, a burgeoning area of study in the network field[9]-[11]. Moreover, recoverability has garnered attention earlier within specific fields of research (e.g., power system reliability[12]).

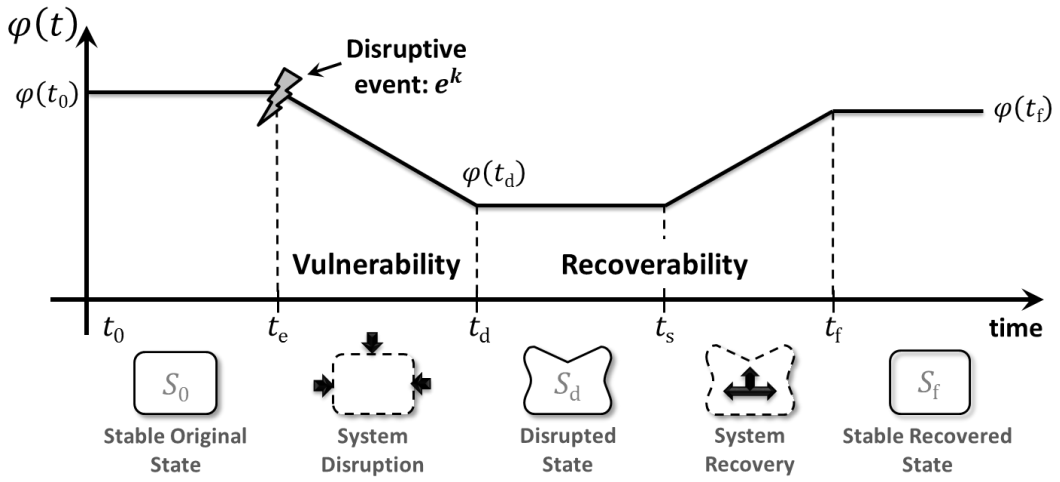


Figure 1. Illustration of network performance, $\varphi(t)$, across different transition states.

With this approach in mind, the evaluation and quantification of these resilience dimensions is possible in a way that is generalizable across many problem instances. Graph encoding and network formulation are often relied upon for applying optimization approaches to a particular system, and it is assumed that networks of interest in this study lend themselves to such modeling paradigms. In previous research, efforts to quantify network characteristics were directed towards graph-theoretic measures (e.g., edge betweenness, centrality). However, performance-driven measures may be of more use in the context of network resilience for decision making purposes[13],[14]. These metrics connect the idea of vulnerability with network flow as a proxy of system performance. As such, node and/or arc importance is a function of the degree to which overall network performance depends on the existence of, capacity of, and flow along that node/arc. This study focuses on the dimension of vulnerability—specifically, the degree to which vulnerability (in terms of performance loss) can be

mitigated by employing an effective defense strategy against probable disruptions with known parameters.

Given that a planner has some prior knowledge that a network faces a disruptive event with uncertainty, it is assumed that the planner will attempt to insulate, fortify, or otherwise harden the network in a way that minimizes the extent of the disruption (vulnerability reduction). Such a defense strategy would incur some cost to implement. The general approach for this study is to employ a defense strategy at a minimal cost that also minimizes network vulnerability. Prior effort has formalized this multi-objective problem[15], taking into account discrete, diverse "attack" scenarios and offering a solution approach for approximating Pareto-optimality to define an overall robust defense strategy. This study makes use of this approach, extending it for multi-commodity networks. The Pareto-optimal defense strategies are specific to a particular attack (hereafter more generally referred to as "disruption"). To explore strategies that are robust to multiple disruptions, the Pareto-optimal frontiers could be consolidated based on stakeholder opinions of the trade-offs between several criteria, including vulnerability reduction across several disruptions and cost. This study uses a multicriteria decision analysis technique, the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), to address these trade-offs, especially given the possibility of a large, high-dimensional Pareto set to consider.

The goal of this paper is to propose a methodology for making robust decisions for reducing vulnerability in multi-commodity networks under uncertain disruptions. The remainder of this paper is as follows. Section 2 describes the proposed methodology.

Section 3 illustrates the methodology with a Swedish rail case study, and Section 4 offers concluding remarks.

Chapter 2: Proposed Methodology

This section discusses the proposed methodology for making robust decisions for reducing vulnerability in multi-commodity networks under uncertain disruptions.

2.1. Single Commodity Formulation

The network vulnerability reduction formulation proposed here extends that which was given previously for a single commodity[15], described as follows.

Let a network be represented by $G = (N, A)$, where N represents the set of nodes (with source node s and sink node t), and A represents the set of links (or edges) between nodes. The capacity of link (i, j) directed from node i to node j is $q_{ij}(a_{ij})$, where a_{ij} is a binary indicator of disruption equal to 1 if the link is disrupted and 0 if the link is not disrupted. It is assumed that if link (i, j) experiences a disruption, $q_{ij}(1) \leq q_{ij}(0)$. The set of (disrupted) capacities across all links is noted as the vector $\mathbf{q}$.

The original formulation considers a set of resources belonging to an adversary divided amongst disruptive events $\mathbf{e}^k \in D$, which further divide those resources so that e_{ij}^k refers to the amount of resources dedicated to disrupt link (i, j) for event k. The set of all disruptive events is D.

The network defender is assumed to be aware of possible disruption scenarios, D, but not aware of the specific components and their locality. The defender employs defense strategy $\mathbf{h}^l$ to minimize the vulnerability of the network to disruption $\mathbf{e}^k$, where h_{ij}^l denotes the resources dedicated to mitigate damage to link (i, j) for strategy l.

Linking disruption and defense strategies with the notion of vulnerability is a contest function found in Eq. (1) based on work by Skapderas[16] and supported by the competing resource strategy by Levitin and Hausken[17]. That is, given disruption k and defense strategy l, the disruptive threat to link (i,j) is the probability that the link's capacity is reduced to zero, represented with $u_{ij}(\mathbf{e}^k, \mathbf{h}^l)$. The exponent m describes contest intensity (which defaults to a value of 1). Note that this contest function is particularly used for attacker/defender scenarios, though it is considered more generally here for disruptions beyond only malevolent attacks where e_{ij}^k could broadly be interpreted as the strength of disruption to link (i,j) and where h_{ij}^l could be a similarly scaled measure of the strength of defense of link (i,j).

$$u_{ij}(\mathbf{e}^k, \mathbf{h}^l) = \begin{cases} \dfrac{\left(e_{ij}^k\right)^m}{\left(e_{ij}^k\right)^m + \left(h_{ij}^l\right)^m} & \text{if } \left(e_{ij}^k\right)^m > 0 \\ 0 & \text{if } \left(e_{ij}^k\right)^m = 0 \end{cases} \tag{1}$$

When $\mathbf{e}^k$ and $\mathbf{h}^l$ are known, each link's survival probability is assumed to be a random variable with probabilities given by Eq. (2).

$$P\left(q_{ij}\left(a_{ij}\right)\right) = \begin{cases} 1 - u_{ij}(\mathbf{e}^k, \mathbf{h}^l) & \text{if } a_{ij} = 1 \\ 0 & \text{if } a_{ij} = 0 \end{cases} \tag{2}$$

Expected network performance, where φ is defined as source-to-sink flow, can be described as $\varepsilon(h^l, e^k) = E[\varphi(\mathbf{q})|\mathbf{e}^k, \mathbf{h}^l]$. And network performance is a function of link

flow, $f(q_{ij}) \in \left(0, q_{ij}(1)\right)$. From these definitions, the formulation is defined as follows. The objectives in Eqs. (3) and (4) are expected network performance and cost of vulnerability reduction, respectively. Constraint (5) is through-network flow balance, constraint (6) is terminal (source-sink) flow balance, constraint (7) enforces link capacity, and constraint (8) ensures non-negativity. These constraints are typical in the maximum flow literature[18]. Note that this formulation accounts for a single source and a single sink for each commodity, which could be imposed as "super source" and "super sink" nodes.

$$\max_{l} \varepsilon(\boldsymbol{h}^l, \boldsymbol{e}^k) \quad \forall\, \mathbf{e}^k \in D \tag{3}$$

$$\min_{l} C(\mathbf{h}^l) \tag{4}$$

s.t.
$$\sum_{i|h_{ij}} f(q_{ij}) - \sum_{k|h_{jk}} f(q_{jk}) = 0 \;\; \forall \;\; i,j \in N, \notin \{s,t\} \tag{5}$$

$$\sum_{j|h_{sj}} f(q_{sj}) - \sum_{k|h_{kt}} f(q_{kt}) = 0 \;\forall\, j,k \in N, \notin \{s,t\} \tag{6}$$

$$f(i,j) \leq q_{ij}(a_{ij}) \;\forall\, (i,j) \in A \tag{7}$$

$$h_{ij}^l \geq 0 \tag{8}$$

The two objectives are in competition, so a Pareto-optimal set and relevant tradeoff schemes have to be determined for decision support.

2.2 Multi-commodity Extension

This study extends the above formulation to include multiple commodities that do not share the same link capacity resource, where the set of commodities is $V, (v \in V)$. The formulation assumes that a given commodity may have multiple points of supply and demand, so each commodity's set of source and sink nodes is defined as S_v and T_v, respectively. Demand for a commodity at a given node z is noted as d_{vz} (where negative demand denotes supply). Flow of commodity v across link (i, j) is denoted by $f_v(i, j)$.

The new objective functions accounting for multiple commodities are found in Eqs. (9) and (10). Eq. (9) refers to the minimization of the largest fraction of unsatisfied demand across all commodities. Where a common measure of network performance (φ from Figure 1) might be maximum flow for a pair of nodes (or all nodes [Nicholson et al. 2016]) or the amount of demand being met in demand nodes, we consider the minimal greatest fraction of unsatisfied demand across commodities. This allows us to represent network performance with a vulnerability measure, which would likely increase after a disruption (as opposed to the decreasing phenomenon in Figure 1). It is assumed that each commodity has equal economic importance, though alternative importance schemes could easily be represented with convex-sum weighting schemes in Eq. (9). The second objective in Eq. (10) refers to the minimization of the cost of implementing strategy l, $C(\mathbf{h}^l)$. The cost function is taken to be specific to the problem and is left in a general form here.

$$\min_{h_{ij}^l} z_1 = \max_{v \in V} \left(1 - \sum_{s_v \in S} \sum_{j \in N} \frac{f_v(s_v, j)}{\sum_{z \in N} d_{vz}} \right) \tag{9}$$

$$\min_{h_{ij}^l} z_2 = C(\mathbf{h}^l) \tag{10}$$

The constraints for the single commodity formulation in Eqs. (3)-(8) are reformulated as Eqs. (11)-(14) to account for multiple commodities and the possibility of multiple sources and sinks for each commodity. Here, Eq. (11) refers to flow balance (amount in equals amount out, less the amount demanded at that node), Eq. (12) refers to commodity flow conservation (no commodity is lost in the network), Eq. (13) refers to the post-disruption capacity constraint on each link, and Eq. (14) refers to the non-negativity condition of the defense strategy (no benefit can come from decreasing resources from a link).

$$\sum_{i|h_{ij}} f_v(i,j) - \sum_{k|h_{jk}} f_v(j,k) - d_{vj} = 0 \; \forall \; v \in V, \; j \notin \{S_v \cup T_v\} \tag{11}$$

$$\sum_{j|h_{s_v j}} f_v(s_v, j) - \sum_{k|h_{k t_v}} f_v(k, t_v) = 0 \;\; \forall \; v \in V \tag{12}$$

$$\sum_{v \in V} f_v(i,j) \leq q_{ijv}(a_{ij}) \; \forall \; (i,j) \in A, \; v \in V \tag{13}$$

$$h_{ij}^l \geq 0 \tag{14}$$

Given that flow across a node is constrained by capacity under probabilistic disruption, the expected value of capacity is represented by Eq. (15).

$$E[\mathbf{e}^k] = \begin{cases} q_{ij}(a_{ij})\left(1 - u_{ij}(\mathbf{e}^k, \mathbf{h}^l)\right) & \text{if } \left(e_{ij}^k\right)^m > 0 \\ q_{ij}(a_{ij}) & \text{if } \left(e_{ij}^k\right)^m = 0 \end{cases} \quad (15)$$

2.2.1 Pareto Optimality

Given the competing objectives, it is necessary to estimate the Pareto set. The exact Pareto set H^* is defined as the set of all non-dominated defense strategies. Of two feasible strategies for reducing vulnerability, $\mathbf{h}^l$ and $\mathbf{h}^{l'}$, given disruption $\mathbf{e}^k$, $\mathbf{h}^{l'}$ dominates $\mathbf{h}^l$ if $\mathbf{h}^{l'}$ outperforms $\mathbf{h}^l$ in at least one objective while $\mathbf{h}^{l'}$ performs at least as well as $\mathbf{h}^l$ in the other objectives. If there exists no $\mathbf{h}^{l'}$ that dominates $\mathbf{h}^l$, then $\mathbf{h}^l$ is non-dominated and $\mathbf{h}^l \in H^*$. Specifically for the bi-objective problem discussed here, a non-dominated strategy essentially refers to a strategy that improves, say, z_1 relative to another strategy but where z_2 degrades relative to the other strategy (and vice versa).

Since $\mathbf{h}^l \in \mathbb{R}^+$, defining H^* precisely, like most multi-objective problems, is at worst, impossible, and at best, computationally difficult. This study proposes, instead, that the NSGA-II heuristic be used to approximate the Pareto set.

To generate the Pareto-optimal frontier (tradeoff space), a heuristic approach to estimating the true Pareto set is favored—namely, the Non-dominated Sorting Genetic Algorithm II (NSGA-II)[19]. This algorithm has been shown to perform favorably both in terms of accuracy (verified by test sets with known Pareto subsets) and in terms of computational complexity for high-dimensional decision spaces in network

formulations. Examples of such problems include multi-objective supply chain problems[20], supply chain resilience[21], stochastic computer network reliability[22], and many other problems based on network and/or graph constructs.

2.3 Robustness Evaluation

The Pareto set is a set of defense strategies (i.e., an investment in protecting a set of links), and each disruption scenario generates one such set. Each set of defense strategies is evaluated against all other disruption scenarios in terms of reduction of the maximum flow through the network and in terms of commodity demand satisfaction (a measure of individual commodity flow). From this set of globally-evaluated solutions, a decision maker assesses the performance of each strategy relative to the commodity flow performance (z_1) for each disruption in D and the overall cost (z_2) of the defense strategy. As the number of disruptions increases, the complexity of the solution set increases. To navigate this complexity, this study utilizes a multicriteria decision analysis technique, TOPSIS, to define a ranking of candidate solutions, where the criteria represent (i) the commodity flow performance for each disruption and (ii) the cost of the defense strategy.

The first step of TOPSIS is to ensure that the criteria being compared are commensurate. This is typically achieved by normalization (or standardization). Range normalization is computationally simple and may offer greater understandability for decision makers in the final solution selection stages, although there are other normalization techniques that may be more appropriate for different networks[23]. Range

normalization is defined on two functional components, one for criteria that are perceived as benefits (advantageous) and one for criteria perceived as costs (disadvantageous). These are shown in Eqs. (16) and (17), respectively. In both cases, a value of one is understood to be the best possible outcome for a given criterion, while a value of zero is the worst.

$$r_{h^l y} = \frac{x_{h^l y} - \min\limits_{h^l \in \mathbf{h}^l} x_{h^l y}}{\max\limits_{h^l \in \mathbf{h}^l} x_{h^l y} - \min\limits_{h^l \in \mathbf{h}^l} x_{h^l y}} \tag{16}$$

$$r_{h^l y} = \frac{\max\limits_{h^l \in \mathbf{h}^l} x_{h^l y} - x_{h^l y}}{\max\limits_{h^l \in \mathbf{h}^l} x_{h^l y} - \min\limits_{h^l \in \mathbf{h}^l} x_{h^l y}} \tag{17}$$

The result is a normalized value, $r_{h^l y}$, denoting the performance of alternative (i.e., defense strategy) h^l for criteria $y \in \{1, \dots, Y\}$, the set of criteria being considered. A weighting scheme can be applied to each criterion, as shown in Eq. (18).

$$b_{h^l y} = r_{h^l y} w_y \tag{18}$$

Since the criteria in this study comprise z_1 and z_2 values from Eqs. (9) and (10), the best solutions seek minimal values in criteria performance. TOPSIS constructs an ideal solution, A^+, componentwise from all considered solutions, selecting the best possible criteria outcomes, as shown in Eq. (19). Similarly, an "anti-ideal" solution, A^-, is constructed componentwise from all worst criteria outcomes, shown in Eq. (20).

$$A^+ = (b_1^+, \dots, b_Y^+), b_y^+ = \min_y b_{h^l y} \tag{19}$$

$$A^- = (b_1^-, \dots, b_Y^-), b_y^- = \max_y b_{h^l y} \tag{20}$$

The solution set is then ordered by comparing each solution to both the ideal and anti-ideal conditions A^+ and A^-. The solutions are ordered by similarity to the ideal condition such that the best solutions have the greatest Euclidean distance from the worst condition (D^-) and the least distance from the best condition (D^+), as described in the similarity metric S^+ in Eq. (21). The solutions with higher S^+ values are considered to be better solutions.

$$S^+ = \frac{D^-}{D^- + D^+} \tag{21}$$

Note that the specific tradeoffs made in the ordering process may lack some transparency for high-dimensional situations, but the ease, speed, and relatability of TOPSIS suggests that it is reasonable for ordering of robust defense solutions in a decision support environment. From the initial solution set, interactive methods for determining stakeholder utility can be employed to inform overall decision support, and these can be incorporated into the TOPSIS methodology as weights, as noted in Eq. (18) above.

2.4 Solution Algorithm

Given that the objective of the modeling process is to understand the impacts of disruption across multiple commodities and to provide decision support from that understanding, this study proposes the following assimilation of the above formulation and techniques.

First, the network is instantiated with nodes, directed links, link capacities for each commodity, and the set of disruption scenarios. These disruption scenarios are constrained by a resource budget, with resources divided equally among links that are targeted. These scenarios must be framed in such a way that the resources directed at the links be commensurate with defense strategy resources. Second, the Pareto-optimal defense strategies are determined for each disruption scenario using the NSGA-II algorithm. These defense strategies are characterized by their objective values: the maximum fraction of satisfied demand and overall strategy cost. Third, the performance of each defense strategy set is weighed against each disruption scenario to determine which defense strategy is most robust across disruptions. The robustness of the defense strategies is characterized by an ordering gained from TOPSIS.

Chapter 3: Illustrative Example: Swedish Rail Network

This algorithmic approach was applied to data for a Swedish railway system of 1,363 stations (nodes) and 1,438 connecting, bidirectional tracks (links), collected from public sources and the infrastructure owner[24]. This system has been studied concerning infrastructure vulnerability in previous publications[24],[25].

3.1. Instrumentation

The data were re-structured and output from Matlab. The problem instance was implemented in the Python programming language using the "networkx" package (version 1.11)[26] to implement and create graph structures and network data structures, the "ecspy" package (version 1.1) to implement evolutionary computations and the NSGA-II evolutionary algorithm, and the Python API for the Gurobi solver platform to solve the single-objective maximum flow sub-problems. All algorithms were performed on a standard laptop computer with a quad-core 2.4GHz processor and 8 gigabytes of memory.

3.2. Network Generation

The freight data comprise 19 different commodities (as shown in Table I) aggregated from publicly available data and train operator data. The data contain granularity issues in that freight movement was consolidated to the level of cargo routes, whereby specific information about individual trains and operators has been removed for sensitivity purposes. As a result, the data lack specific supply and demand values for stations and capacity parameters for links, though this study makes use of the estimation approach

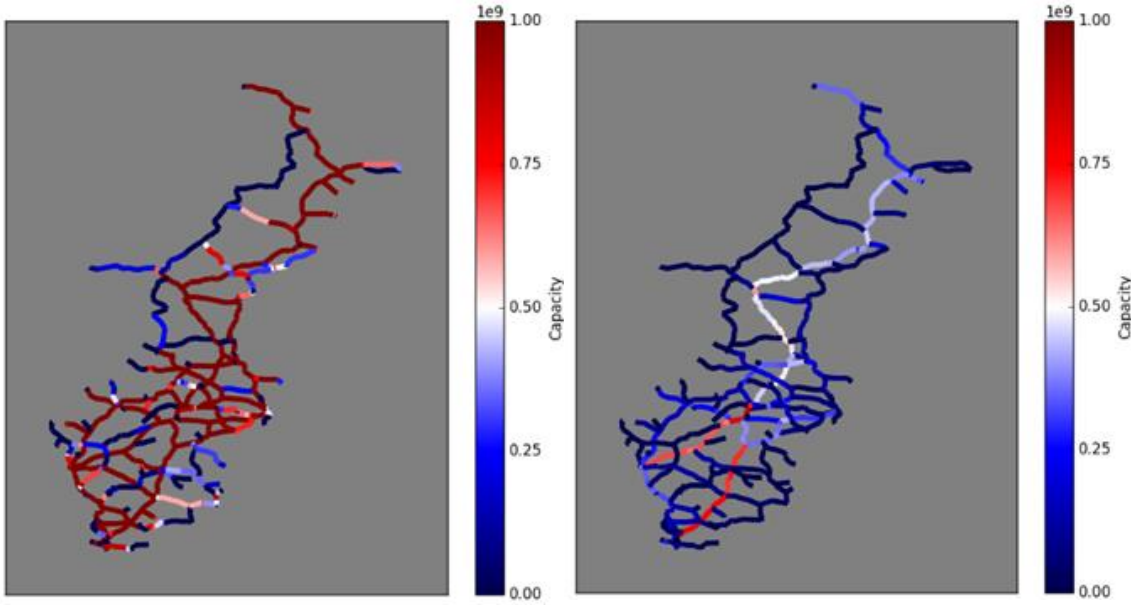
for these parameters from a previous study[27]. Supply and demand values for each commodity were distributed across those stations over which train operators shipped each commodity, with the values being proportional to the number of routes bearing that commodity across that node. Similarly, link capacities were estimated by assessing freight movement in the rail network such that the resulting network has some degree of slack. Table 1 provides the commodity descriptions (translated from Swedish) and estimates for supply and demand. Figure 2 depicts two selected examples of capacitation outcomes from this process, showing that different commodities have different levels of movement through different paths in the network. Conceptually, this could be due to availability of different train car types, or supply-demand interactions.

Given that supply is not equivalent to demand for each commodity, the values are adjusted to the minimum of the two values for the base case of this network. In this way, the best possible performance of the network equates to 100% demand satisfaction.

Generating the network relied heavily on data structures provided by the Python package "networkx" [27]. The final network consisted of the original 1,363 nodes and 2,876 links between them (since each of the original 1,438 links is considered bi-directional).

Table 1. Raw supply and demand data for each commodity (in kilotons).

Index	Commodity	Supply	Demand
1	Agriculture, forest, fishing	228	284
2	Coal, crude oil, natural gas	27	19
3	Ore	210	262
4	Food, beverage, tobacco	281	366
5	Textile, leather	240	262
6	Wood, cork, pulp, paper	245	276
7	Petroleum products	198	217
8	Chemicals, rubber, plastics	186	187
9	Other non-metallic mineral	270	258
10	Fabricated metal products	216	193
11	Machinery and equipment	263	251
12	Transport equipment	240	269
13	Furniture, other manufactured	248	239
14	Return materials and recycling	256	380
15	Post and packages	0	0
16	Equipment for transportation	238	260
17	Moving goods, vehicles for repair	0	0
18	Loader and grouped goods	287	241
19	Unidentifiable goods	293	267

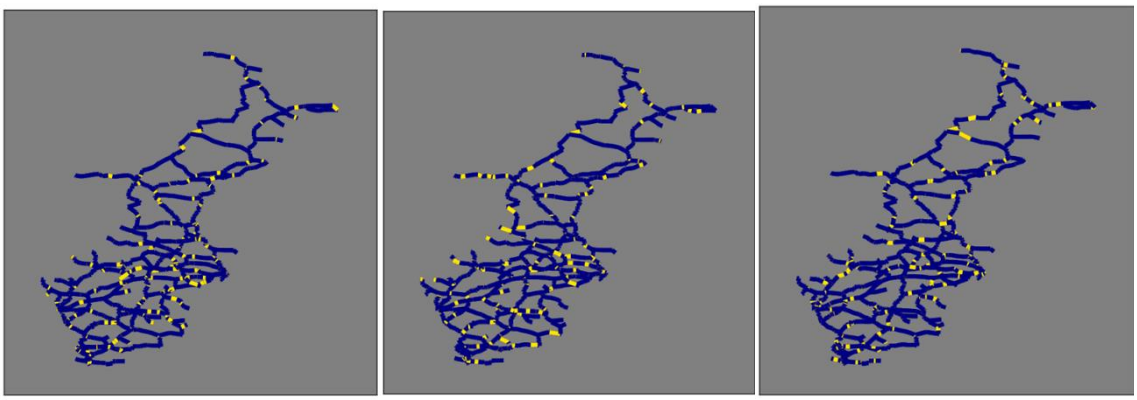


(a) commodity 2 (b) commodity 7

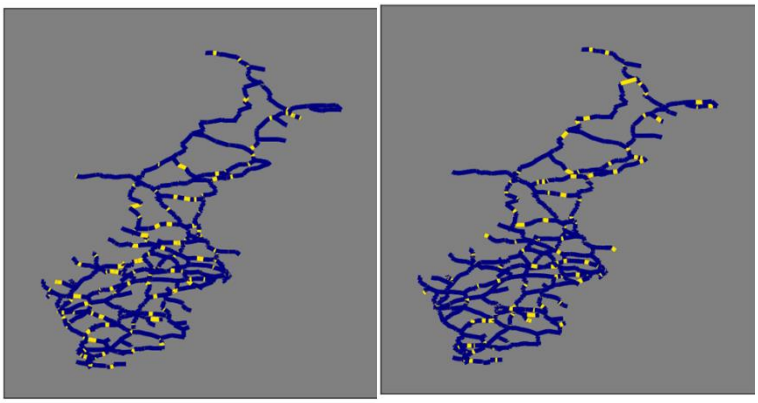
Figure 2. Capacitation of the network for commodities 2 (Coal, crude oil, natural gas) and 7 (Petroleum products).

3.3. Disruption Scenarios

Since there are no disruption data for this problem instance, and since costs for hardening the system are unknown, five separate disruption scenarios were generated. For each scenario, 288 links (10% of total) were selected at random and were allocated 10 units of disruption resource. Lacking a disruption scenario, such a distribution may be realistic in the event of a Swedish winter with extremely heavy snowfall combined with hard winds. Thus, simulating the probabilities of failure over five disruption scenarios should help decision makers to determine the defense strategy (e.g., investing in track clearance resources) that is most robust to this problem. An example disruption of the network is shown in Figure 3.



(a) scenario 1 (b) scenario 2 (c) scenario 3



(d) scenario 4 (e) scenario 5

Figure 3. The five disruption scenarios for the network, with yellow links denoting full disruption.

3.4. Pareto Frontier Estimation

The determination of Pareto frontiers was approached for each disruption scenario individually.

To initialize the NSGA-II algorithm, a set of starting solutions is necessary. Each solution is a list of defense resources allocated to corresponding links of the network. In trial runs and preliminary tuning of the NSGA-II algorithm, different methods were used to randomly generate high-performing, sufficiently diverse initial solution sets, but it was found that a null set, together with a high mutation rate, performed better than the other methods. The set of solutions—or evolutionary "population"—consisted of 50 members.

Because of the size of the network and the limitations of computational resources, parameter tuning was a critical step to ensure high-quality estimates. Given the 50-

member population, a mutation rate of 0.8 converged to relatively good solutions after about 10 generations, so a final value of 20 generations was chosen, pushing the computation resources to refine the Pareto estimate as much as possible. Because the number of generations is limited, it was found that the combination of a high mutation rate with the internal greedy mechanisms of NSGA-II struck a good balance between exploration of the solution space and exploitation of superior solutions.

With the problem fully instantiated, the NSGA-II iterations (generations) began. An iteration consisted of an application of the contest function to each link for each solution in the population given the selected disruption scenario and each solution in the population. The outcome of the contest function is a probability value, which was assessed with a pseudo-random number generator. As described in Eq. (15), the probabilistic outcome is binary, and if the link was found to be "disrupted," the capacity for all commodities on that link was reduced to 0. Likewise, if the outcome of the contest function is "not disrupted," the commodity capacities are unchanged. A new, separate, disrupted network is then created for each h^l in the solution set, and the minimal value of the greatest fraction of unsatisfied demand (z_1) is calculated through solving a multiple-commodity network flow problem (for each h^l) implemented in the Python API for the Gurobi linear programming platform. The fitness of each solution in the population was determined from the z_1 value and the cost of the solution, taken to be the sum of the defense resources allocated to each link (z_2).

NSGA-II was then allowed to iterate for 20 generations using blend crossovers and the canonical NSGA-II selection and diversity methods to produce an estimate of the Pareto set of best-performing defense strategies.

3.5. Robustness Evaluation and Strategy Ranking

The final solution set obtained from each disruption scenario was then assessed for performance in each of the other five disruption scenarios. This resulted in a final solution set with a total of 250 solutions (population size multiplied by number of disruptions). Each of these solutions generated a disrupted graph instance for each of the five disruptions, and the resulting 1,250 graphs were assessed for z_1, while z_2 was constant across each of the 250 solutions.

Six criteria were considered: thez_1 outcome from each of the five disruption scenarios and overall cost, z_2. for the TOPSIS ranking. The top-ranked of the 250 defense strategies are provided in Table 2.

Table 2. TOPSIS rankings for top 12 defense strategies.

Defense population	Population member	S+	Rank	Vulnerability z_1 for scenario k					Cost z_2
				$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	
1	0	0.462	1	0.099	0.400	0.494	0.619	0.665	14294
1	2	0.459	2	0.238	0.412	0.549	0.627	0.666	13943
1	3	0.443	3	0.257	0.427	0.559	0.665	0.673	13927
1	1	0.419	4	0.213	0.409	0.549	0.627	0.665	14286
1	7	0.415	5	0.323	0.452	0.580	0.665	0.677	13944
1	5	0.404	6	0.276	0.445	0.579	0.665	0.673	14110
1	6	0.401	7	0.323	0.449	0.579	0.665	0.677	14038
1	12	0.390	8	0.361	0.469	0.586	0.665	0.712	13980
4	3	0.387	9	0.716	0.716	0.716	0.273	0.716	13773
1	11	0.387	10	0.355	0.469	0.586	0.665	0.712	14009
3	2	0.383	11	0.716	0.716	0.198	0.716	0.716	13938
1	17	0.382	12	0.361	0.474	0.591	0.665	0.716	14018

Mentioned previously, running NSGA-II for each disruption scenario generated 50 Pareto-optimal solutions (250 total). Between the crossover operator and the mutation operator used in the genetic iterations of NSGA-II, most solutions allocated defense resources to links that were not disrupted in the scenario. This was initially unexpected, but in the scheme of multiple disruptions, distributing resources with a broader brush conferred some degree of robustness to the solution sets, as the distribution of resources was appreciably uniform for non-disrupted links. This robustness creates inefficiency in solution costs and overall costs are inflated compared to parsimonious distribution of defense resources only to those links in the disruption set. It is expected, however, that longer run-times of NSGA-II over more disruption scenarios would improve the Pareto estimation and reduce this artificial inflation, confining resource allocation to the most vulnerable links.

Despite these inefficacies, the quality of the solution set in a decision support environment remains high. Given the real-world political and socio-economic complexities of implementing defense strategies, a number of defense strategies with similar performance and costs but different allocations of resources might be desired. Perhaps, in the above example, resource allocation is limited in certain regions because of logistics, legislated spending caps, or some other difficult-to-model reality. Moreover, defense strategies that augment all links to some extent and critical links to a greater extent are defensibly realistic for certain probabilistic disruption scenarios. For example, those implementations, depending on the system of interest, might manifest as railway bridge retrofitting for spring floods, sandbagging flooding rivers, or adding

parallel tracks. In a decision support environment, a decision maker may appreciate alternatives that, although not Pareto-optimal by definition, still confer non-trivial vulnerability reduction and overall increased system resilience. Further, TOPSIS offers a fast, transparent, easily-understandable ordering of these solutions. The distance-driven approach helps identify the superior solutions regardless of the quality of Pareto-optimality estimation from the NSGA-II output.

Chapter 4: Concluding Remarks

This study offers a formulation and modeling approach to assessing a multiple-commodity system's vulnerability to disruption events. The system is abstracted to a graph representation of a network of nodes and directed, capacitated links. The network formulation aims to reduce the cost of a defense strategy while maintaining a high degree of demand satisfaction for each commodity. These objectives are based on the decision to allocate defense resources to specific links in the network. A heuristic search by a well-established genetic algorithm estimates the Pareto frontier for each disruption scenario in the set of disruption events. These Pareto frontiers form the solution set and are incorporated into a decision support environment with TOPSIS. From TOPSIS, the criteria of each solution (i.e., total cost and, for each disruption, demand satisfaction) are compared to an ideal condition and the solutions are ordered in terms of robustness of each defense strategy to all possible disruptions. The solution method allows for some decision maker interaction to account for real-world difficulties of implementing specific defense strategies (e.g., weight given to disruptions assumed to be more likely or of bigger concern).

4.1. Limitations

This approach has potential to be useful in several ways, but the consideration of a solution's response to each disruption after the heuristic estimation limits its ability to be robust to the other disruptions. This could be avoided by adding a demand satisfaction objective for each disruption (like z_1), but it is anticipated that this would prohibitively increase computation time for this problem instance and implementation

given NSGA-II's computational complexity and the data structures employed in the algorithm.

Another issue was that the network is fairly sparse. The sparsity of the underlying graph causes significant problems when assessing impacts of the disruption strategy we chose. It was found that if a disruption occurred on a sparse branch or subgraph then the solution quality would be highly dependent on the randomness of NSGA-II variator operator to "find" that branch. That is, if no solutions evolved to address the disruption of the link in that sparse subgraph, then the solution would not mitigate the disruption of that link. As a result, the demand satisfaction objective (z_1) would suffer and appear "frozen" for a given disruption across multiple population members. This can be seen in Table II where the z_1 objective value for a disruption is the same for multiple solutions. Sparsity of the network has similar consequences for the robustness evaluation across the different disruptions, as seen in the dominance of disruption scenario 1 in Table 2. It was found that the particular disruptions generated in disruption scenarios 2 through 5 had several disrupted links in sparse subgraphs. Again, the exploration mechanics of NSGA-II had difficulty "finding" these links in order to allocate defense resources to them. Disruption scenario 1 had fewer disruptions on sparse subgraphs. The result is the coincidental dominance of defense strategy 1 (which was generated from disruption scenario 1). To overcome issues associated with sparsity, it may be useful to employ graph-reducing algorithms. This might cause difficulties with capacitation, so any such reduction or simplification of the graph will need to account for bottlenecking of capacity along a sparse subgraph after disruption. That is, reducing sparsity in the graph

is secondary to retaining integrity of the disruptive event and defensive strategy; over-simplification may reduce the effective meaning of a given solution.

A further limitation is that this model assumes deterministic supply and demand values, and that supply and demand should be equivalent. Given the model's high complexity under assumptions of determinism, incorporating stochasticity may make the model prohibitively complex to solve in realistic time.

Finally, the complexity of the data structures and the slightly long solution times (tens of hours) may limit the applicability of this approach to larger or more time-sensitive problems, though it is argued that this approach should be used for longer term planning where run time is not likely an issue. As discussed above, simplification of the network may avoid these computational disadvantages, but would require careful thought for each specific problem instantiation.

Despite these limitations, the proposed approach offers robust solutions in a decision support environment to begin addressing network vulnerability to a certain kind of disruption. Importantly, no single outcome from the TOPSIS ranking should be treated as a superior solution, but rather as a starting point to approach real-world complexities associated with the system. Note that TOPSIS is just one of several techniques that could be chosen to compare discrete strategies under multiple criteria. TOPSIS was chosen here due to its simplicity and its ability to implement a compromise solution.

The choice of decision analysis technique could influence the ranking of strategies[28], though a comparison is not sought here.

4.2. Future Work

Future work may entail performing a computational performance analysis for several types of networks across different evolutionary heuristics. That is, NSGA-II may be superior in some regards, but more recent heuristics may be better-suited to dealing with complex data structures that often arise in network modeling with multiple commodities. Along these lines, network simplification and sparsity-reducing algorithms might be explored to reduce the number of non-contributing nodes and links of a real-world network. Eliminating even a few nodes and/or links could significantly reduce computation times and encourage the use of more evolutionary generations to explore and exploit useful defense solution attributes.

Within the resilience modeling framework, future research will elucidate trade-offs between vulnerability reduction and recoverability improvement given disruptions with some stochastic component (e.g., trade-off between pre-disruption and post-disruption investments).

References

[1] The White House, Office of the Press Secretary. 2013. *Presidential Policy Directive/PPD-21: Critical Infrastructure Security and Resilience.*

[2] Turnquist, M. and E. Vugrin. 2013. Design for Resilience in Infrastructure Distribution Networks. *Environment Systems and Decisions*, **33**(1): 104-120.

[3] Hosseini, S., K. Barker, and J.E. Ramirez-Marquez. 2016. A Review of Definitions and Measures of System Resilience. *Reliability Engineering and System Safety,* **145**: 47-61.

[4] Henry, D. and J.E. Ramirez-Marquez. 2012. Generic Metrics and Quantitative Approaches for System Resilience as a Function of Time. *Reliability Engineering and System Safety*, **99**: 114-122.

[5] Barker, K., J.E. Ramirez-Marquez, and C.M. Rocco. 2013. Resilience-Based Network Component Importance Measures. *Reliability Engineering and System Safety*, **117**: 89-97.

[6] Pant, R., K. Barker, J.E. Ramirez-Marquez, and C.M. Rocco. 2014. Stochastic Measures of Resilience and their Application to Container Terminals. *Computers and Industrial Engineering*, **70**: 183-194.

[7] Holmgren, A.J. 2006. Using Graph Models to Analyze the Vulnerability of Electric Power Networks. *Risk Analysis,* **26**(4): 955-969.

[8] Johansson, J., H. Hassel, and E. Zio. 2013. Reliability and Vulnerability Analyses of Critical Infrastructures: Comparing Two Approaches in the Context of Power Systems. *Reliability Engineering and System Safety*, **120**: 27-38.

[9] Nurre, S.G., B. Cavdaroglu, J.E. Mitchell, T.C. Sharkey, and W.A. Wallace. 2012. Restoring Infrastructure Systems: An Integrated Network Design and Scheduling Problem. *European Journal of Operational Research,* **223**(3): 794-806.

[10] Gonzalez, A., L. Duenas-Osorio, M. Sanchez-Silva, and A.L. Medaglia. 2016. The Interdependent Network Design Problem for Optimal Infrastructure System Restoration. *Computer-Aided Civil and Infrastructure Engineering,* **31**(5): 334-350.

[11] Almoghathawi, Y., K. Barker, and L.A. McLay. 2016. Resilience-Driven Restoration Model for Interdependent Infrastructure Networks. Under review in *Annals of Operations Research*.

[12] Wang, P. and R. Billinton. 2002. Reliability Cost/worth Assessment of Distribution Systems Incorporating Time-varying Weather Conditions and Restoration Resources. *IEEE Transactions on Power Delivery*, **17**(1): 260-265.

[13] LaRocca, S., J. Johansson, H. Hassel, and S. Guikema. 2014. Topological Performance Measures as Surrogates for Physical Flow Models for Risk and Vulnerability Analysis for Electric Power Systems. *Risk Analysis*, **35**(4): 608-623.

[14] Nicholson, C.D., K. Barker, and J.E. Ramirez-Marquez. 2016. Flow-Based Vulnerability Measures for Network Component Importance: Experimentation with Preparedness Planning. *Reliability Engineering and System Safety*, **145**: 62-73.

[15] Ramirez-Marquez, J.E., C.M. Rocco, and K. Barker. 2016. Bi-Objective Vulnerability Reduction Formulation for a Network Under Diverse Attacks. Under review in *Journal of Risk and Uncertainty in Engineering Systems*.

[16] Skapderas, S. 1996. Contest Success Functions. *Economic Theory,* **7**(2): 283-290.

[17] Levitin, G. and K. Hausken. 2008. Protection vs. Redundancy in Homogenous Parallel Systems. *Reliability Engineering and System Safety,* **93**(10): 1444-1451.

[18] Ahuja, R.K., T.L. Magnanti, and J.B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Upper Saddle River, New Jersey.

[19] Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation,* **6**(2): 182-197.

[20] Bandyopadhyay, S. and R. Bhattacharya. 2014. Solving a Tri-Objective Supply Chain Problem with Modified NSGA-II Algorithm. *Journal of Manufacturing Systems,* **33**(1): 41-50.

[21] Dixit, V., N. Seshadrinath, M.K. Tiwari. 2016. Performance Measures Based Optimization of Supply Chain Network Resilience: A NSGA-II + Co-Kriging Approach. *Computers and Industrial Engineering*, **93**: 205-214.

[22] Lin, Y.-K. and C.-T. Yeh. 2012. Multi-Objective Optimization for Stochastic Computer Networks Using NSGA-II and TOPSIS. *European Journal of Operational Research,* **218**(3): 735-746.

[23] Chakraborty, S. and C.-H. Yeh. 2009. A simulation comparison of normalization procedures for TOPSIS. *Proceedings of the International Conference Computers & Industrial Engineering*.

[24] Svegrup, L. and J. Johansson. 2015. Vulnerability Analyses of Interdependent Critical Infrastructures: Case Study of the Swedish National Power Transmission and Railway System. *ESREL 2015*, Zürich, Switzerland.

[25] Johansson, J., H. Hassel, and A. Cedergren. 2011. Vulnerability Analysis of Interdependent Critical Infrastructures: Case Study of the Swedish Railway System. *International Journal of Critical Infrastructures*, **7**(4): 289-316.

[26] Hagberg, A.A., D.A. Schult, and P.J. Swart. 2008. Exploring Network Structure, Dynamics, and Function using NetworkX. *Proceedings of the 7th Python in Science Conference*, Pasadena, CA.

[27] Whitman, M.G., K. Barker, J. Johansson, and M. Darayi. 2016. Component Importance Measures for Multi-Commodity Networks: Application in Swedish Railway. Under review in *Networks and Spatial Economics*.

[28] Opricovic, S. and G.H. Tzeng. 2004. Compromise Solution by MCDM Methods: A Comparative Analysis of VIKOR and TOPSIS. *European Journal of Operational Research*, **156**(2): 445-455.