APPLICATION OF MODEL PREDICTIVE CONTROL TO A GEOTHERMALLY HEATED BRIDGE DECK

By

FENG XIE

Bachelor of Science
Zhejiang University
Hangzhou, China
1997

Master of Science
Zhejiang University
Hanghzou, China
2000

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the Degree of
MASTER OF SCIENCE
December, 2004

APPLICATION OF MODEL PREDICTIVE CONTROL TO A GEOTHERMALLY HEATED BRIDGE DECK

Thesis Approved:

James R Whiteley
Thesis Advisor

Karen A. High

R. Russell Rhinehart

J. D. Spitler

A. Gordon Emslie
Dean of the Graduate College

ACKNOWLEDGEMENTS

I wish to thank Dr. James (Rob) Whiteley for his guidance and inspiration of this research. I would like to thank Mr. Derek (Deke) Arndt for making weather information I requested available. I would also like to thank Mr. Xiaobing Liu for the bridge deck model technical support he offered.

Special thanks to my wife, Yan Lin, for her constant encouragement and support.

Finally, I want to thank the Federal Highway Administration of U.S. Department of Transportation, the State of Oklahoma and the School of Chemical Engineering of Oklahoma State University for their financial support.

TABLE OF CONTENTS

Chapter Page

I. Introduction

II. Background

III. System Implementation

LIST OF TABLES

LIST OF FIGURES

Figure Page

NOMENCLATURE

$\mathbf{a}$	Step response model
$\mathbf{A}$	Dynamic Matrix
A/D	Analog to Digital
CV	Controlled Variable
D/A	Digital to Analog
DMC	Dynamic Matrix Control
DV	Disturbance Variable
$\hat{\mathbf{e}}$	Projected error vector
GMT	Greenwich Mean Time
GUI	Graphical User Interface
HBT	Heated Bridge Technology
i	iteration count
m	Control horizon
MPC	Model Predictive Control
MV	Manipulated Variable
n	Length of step response model
NOAA	National Oceanic and Aeronautic Administration
OCS	Oklahoma Climatological Survey
OSU	Oklahoma State University
OU	University of Oklahoma
p	Prediction horizon
PC	Personal Computer
PetLab	Petroleum Lab (Oklahoma State University)
QDMC	Quadratic Dynamic Matrix Control
$\hat{r}$	Reference trajectory
RUC	Rapid Update Cycle
t	time
$\mathbf{u}$	Vector of MV moves
u	MV moves
$\Delta\mathbf{u}$	Vector of MV adjustments
$\Delta\mathbf{u}^{0}$	Initial guess of MV adjustments
$\Delta\mathbf{u}^{1}$	New MV adjustments
$\Delta\mathbf{u}^{*}$	Optimized MV adjustments
$u_{\min}$	Minimum MV constraint (°C)
$u_{\max}$	Maximum MV constraint (°C)
WAN	Wide Area Network
$\hat{\mathrm{y}}$	Predicted average bridge deck surface temperature response with no future control action (°C)

y	Bridge surface temperature (°C)
Φ	Objective function
$\boldsymbol{\Gamma}$	Output error weighting matrix
$\boldsymbol{\gamma}$	Vector of output error weights
$\boldsymbol{\Lambda}$	Input weighting matrix
$\boldsymbol{\lambda}$	Vector of input weights

CHAPTER I

INTRODUCTION

1.1 Project Summary

Accumulation of ice or snow on highways, especially on bridges in the winter is dangerous. It can cause serious economic and safety impact to people and the nation. Traditional methods to remove ice and snow have been to use salt or other deicing chemicals to suppress the freezing point and prevent ice formation. However, there are two major problems associated with this method. First is a reduction in bridge life due to the corrosive effects of salt on rebar and other structural steel. The second is the environmental impact when salt runs off into water bodies.

Recently, heated bridge technologies (HBT) have been studied as a promising, effective and economic alternative, which can overcome the disadvantage of traditional anti-icing methods [3] [13] [17] [20] [23] [24]. The main idea of the HBT is, with a deck heating system installed inside the bridge, the bridge deck temperature can be kept above freezing point when an icing/snow threat is detected. Therefore, the snow accumulation and ice can be prevented.

Because icing conditions may be infrequent, a control system is needed to determine when the heating system should be engaged. Therefore, energy consumption is reduced. However, the control systems used in the previous testing heated bridges [17] are not effective [13] in preventing the initial accumulation of ice or snow. A new control algorithm needs to be integrated.

The work described in this thesis details the implementation of an autonomous control system to operate a geothermal heated bridge. Geothermal heated bridge means the bridge is heated by the energy stored in the earth.

The development of the control system is an integral part of the Oklahoma State University (OSU) Geothermal Smart Bridge project. The project's goal is to "research, design, and demonstrate technically feasible, economically acceptable, and environmentally compatible Smart Bridge systems to enhance the nation's highway safety and to reduce its life cycle cost" [20]. The project is funded by the Federal Highway Administration of U.S. Department of Transportation and the State of Oklahoma. People from the departments of Mechanical Engineering, Chemical Engineering and Civil Engineering, OSU and personnel from Oklahoma Climatological Survey at the University of Oklahoma have collaborated on the project.

Building on the previous research [3] [13], this work completes the implementation of a Model Predictive Control (MPC) based control system for the Smart Bridge. Through the real time tests in winter 2003-2004, the control system has demonstrated its ability to pre-heat the bridge ahead of the icing threats and keep the bridge ice free. Additionally, the controller software is objective oriented and highly modularized. It can be used as a framework for future control system developments.

1.2 Thesis Organization

The remainder of this thesis is organized as follows.

In Chapter II, all the background related to this work is introduced. First, the snow/ice problem is detailed. Then current technologies used for anti-icing are discussed. Three Heat Bridge Technologies are compared. After the analysis of existing heated bridge control systems, the OSU Geothermal Smart Bridge Project is introduced. A brief review of Model Predictive Control is included.

Chapter III provides detailed information how the control system is implemented, including the hardware architecture, software architecture and detailed information of the components of the controller.

In Chapter IV and Chapter V, several case studies are presented. Chapter IV focuses on the real time performance of the control system during the snow events in the winter of 2003-2004. Chapter V describes modifications to enhance the control system performance based on the result of Chapter IV.

Chapter VI provides concluding remarks and recommendations for future research.

CHAPTER II

BACKGROUND

2.1 Snow or Ice Accumulation

Transportation systems are vital to the health of a modern nation's economy. In most countries, the highway system is the lifeline for their business and industry. Therefore, the safety of the highway system is always given a high consideration. Furthermore, because of the huge construction expense of the highway, investors would like to see the highway fully used during its designed life time, in other words, no reduction of the highway's life time.

However, driving on the highway can be dangerous during the winter, early spring and late fall. In some weather conditions like snow, sleet or freezing rain, drivers can easily lose control of their vehicles because of the accumulation of ice and snow on highways, and low visibility conditions. These conditions contributed to 4,180 fatal crashes and 11,638 deaths or injuries in 2002 [8].

The condition is even worse on bridges. Snow or ice accumulation is more likely to occur on bridge surfaces for two reasons. First, with all surfaces exposed to the air, bridge decks lose heat much faster than roads. Therefore, the surface temperature of the bridge can be lower than that of the approaching road. Second, bridges frequently cross watercourses. Since bridges are surrounded by moist air, the surfaces are wetter which increase the possibility of snow or ice accumulation.

Snow or ice accumulation on bridges are hazardous because drivers usually assume the same surface conditions when they approach a bridge from a clear road, even when warning signs are posted. The suddenly changed road surface condition often results in vehicle accidents, injures, and lives lost.

Because of what is described above, preventing or eliminating snow or ice accumulation on bridges will highly increase the safety of the highway system.

2.2 Conventional Method

Currently, the prevalent method to prevent or remove snow or ice accumulation is the application of salt or other deicing chemicals. Salt and these deicing chemicals can suppress the freezing point of water and prevent ice formation.

However, when should the salt or deicing chemicals be spread? How much salt or deicing chemicals should be used? For different weather conditions, what kind of deicing chemicals should be used? Though some researches have given some instructions [14], these questions are still under study. Furthermore, salt must be dispersed repeatedly during heavy snow and is incapable of melting ice during extreme cold weather conditions.

In addition, there are two major problems associated with this method. First is the reduction in bridge life due to the corrosive effects of salt on rebar and other structural steel of the bridge. Bridges are more likely to reach their designed life when no salt is used. This is the major motivation of research for de-icing methods other than salt or chemicals.

The second is the environmental impact. Though many chemicals are declared as environment friendly, no one can assure the long time effect. And the environmental problem is inevitable when salt is used and runs off into water bodies.

2.3 Alternative Method

An effective and economic alternative is to use heated bridge technologies (HBT). With a deck heating system installed, the bridge deck temperature can be kept above freezing point. Therefore, snow will be melted and the accumulation will be eliminated.

There are three types of heated bridge technologies: hydronic, heat pipe and electrical [17]. They use different methods to achieve the same goal, heat the bridge deck.

Hydronic: Heat transfer fluid heated by some heat sources is circulated through pipes installed just below the surface of the bridge pavement. Heat is released to the bridge deck by conduction. The fluid usually is circulated by pumps. The heat sources can be high-level sources such as boilers, or low-energy geothermal sources such as well water or the earth itself.

Heat pipe: A pipe filled with evaporable fluid transfers heat from one end (hot) to the other end (cold) by natural convection. The cold end is installed below the bridge pavement. At the hot end, fluid is vaporized by some heat sources, the same as what used in the hydronic heated bridge system. When the vapor rises to the cold end, it condenses and releases the heat. With the fluid motion, heat is transferred from heat source to the bridge. Heat pipe technology usually does not need pumps.

Electrical: When electrical current flows through a conductor, some of the electrical energy will be converted into heat. This is the principle of the electrical heating technology. Electrical resistance wires are embedded in the bridge deck. They provide the heat to prevent icing.

Each heated bridge technology has its benefits and limitations. Table 2-1 compares the cost and effectiveness of these technologies [17].

	Installation Fee	Operation Fee	Heat Effectiveness
Hydronic	High	Medium	High
Heat pipe	High	Low	Low
Electrical	Medium	High	High

Table 2-1: Heated Bridge Technology Comparisons

Funding for HBT research was provided between 1992 and 1997 as part of the Applied Research and Technology program (Section 6005) of the Intermodal Surface Transportation Efficiency Act [17]. Eight heated bridge decks were constructed in five states as a result of this program. A summary and discussion of these bridges can found in [13].

2.4 Heated Bridge Control System

A bridge deck heating system can be run to keep the bridge's surface temperature above zero at all times. Ideally, the control system should automatically engage the deck heating system only when the deck needs be heated. Therefore, the operation cost can be minimized.

Previous study on the existing heated bridges' control systems showed that they all utilized On/Off control [13]. It means the controller will turn the heating system on if a certain set of criteria is matched and off otherwise. Table 2-2 summarizes the control rules used in the existing heated bridges.

Bridge	**Conditions Required to Turn Heating System ON**	**Conditions Required to Turn Heating System OFF**
Tenth Street Pedestrian Viaduct – Lincoln, Nebraska	Pavement T < 39°F AND Air T < 36°F AND Moisture on Bridge Deck	Pavement Temperature > 55°F
Silver Creek – Salem, Oregon (control rules partially proprietary)	Air T < Specified Value AND Moisture on Bridge Deck	Air T > 35-37°F OR Pavement T > Specified Value
Highland Interchange – Portland, Oregon	20°F < Air T < 33°F AND Dew Point > 0°F	Unknown
Second Street Overcrossing – Hood River, Oregon	Air T < 35°F AND Relative Humidity > 95%	30-minute minimum runtime AND Pavement T > 36°F
U.S. 287 – Amarillo, Texas	Pavement T < 35°F AND Precipitation Forecast	Unknown
Route 60 Bridge – Amherst County, Virginia	Snow or Ice on Pavement OR Precipitation Present AND Air T < 35°F OR Moisture on Bridge Deck AND Pavement T < 35°F	No Moisture on Pavement for 10 minutes OR Pavement T > 40°F

Table 2-2: Rules Used In Existing Heated Bridges

In these rules, the rules to turn on the heating system are most important. They determine when the heating system starts to heat the bridge. However, the heating system will be engaged only when these icing conditions have been detected. Heating bridge deck is a very slow procedure, usually hours, therefore, bridges can not be heated enough in a short time; snow or ice accumulation is inevitable at the beginning of a freezing precipitation event.

Heating a bridge is a very slow procedure, thus there is a need to pre-heat the bridge ahead of the icing events. For example, if the controller knows there is a snow, which expected to arrive six hours later, it can start heating the bridge right now. The above zero temperature condition can be achieved in this way. To act in this fashion, the controller should have the ability to know the weather condition in the future. However, all the existing heated bridge systems use local instruments to get real time weather conditions. They lack the ability to know weather conditions in the future. Fortunately, such kinds of weather information systems are available [1] [2].

2.5 OSU Geothermal Smart Bridge

The OSU Geothermal Smart Bridge project combines state of the art technologies in heated bridge technology, heat pump research, advanced weather forecast, and control strategies. The OSU Smart Bridge utilizes the hydronic heated bridge technology. It makes use of a ground source heat pump system which recovers energy stored in the earth, and uses it to heat fluid that is circulated through the bridge deck [20].

Figure 2-1 shows the major components of the OSU Geothermal Smart Bridge system. The bridge deck has tubes buried in the pavement. Warm fluid is circulated through these tubes and heats the bridge deck by conduction. The fluid is heated by heat pumps, which transfer the energy stored in the earth by heat exchangers.

Heat pump is effective in energy usage. With geothermal energy as a heat source, the OSU smart bridge's operating fee is competitive. In addition, during summer, the heat pump can work reversely, transferring heat from the bridge to the ground. This is another contribution to extend bridge life.

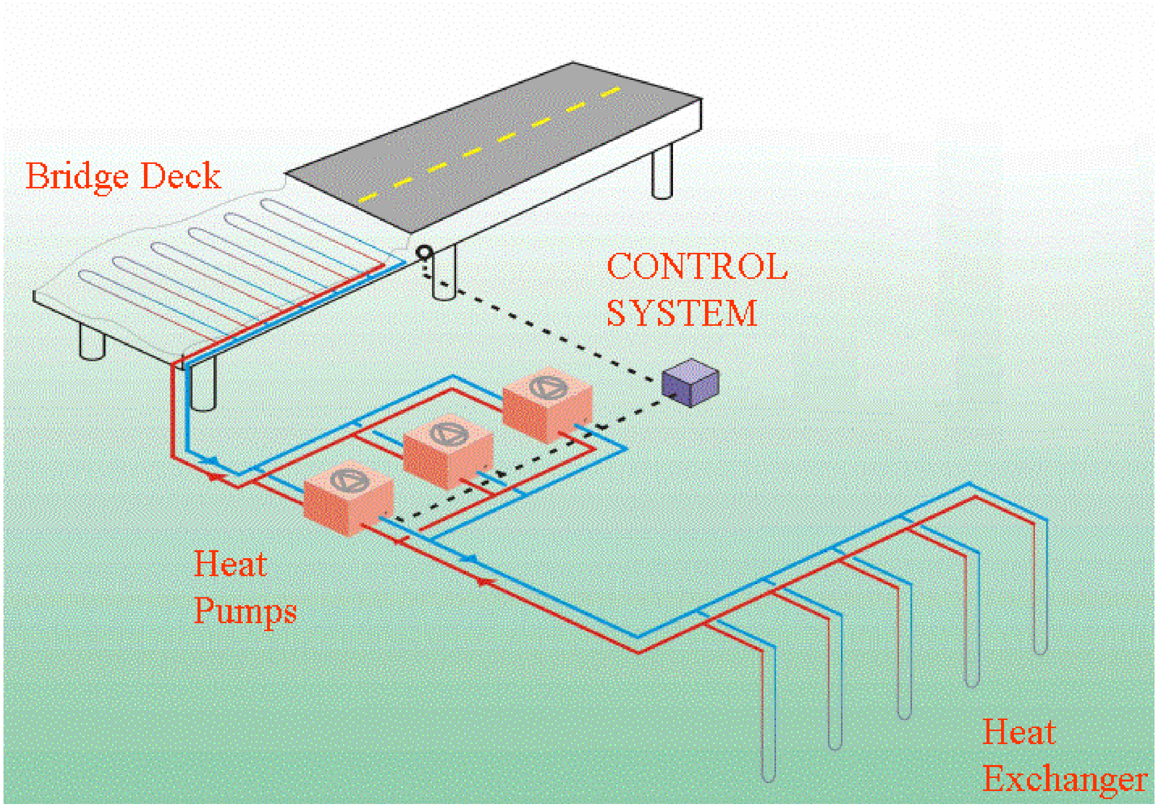


Figure 2–1: OSU Geothermal Smart Bridge System

The bridge heating system equips a control system, which decides when to engage the heat pumps and when to stop them. The decisions are based on the measurements at the bridge site, local weather information and forecasts. The automatic nature of the control has given rise to the informal name "Smart Bridge" [20].

To test and demonstrate the technologies used in the OSU Geothermal Smart Bridge project, a medium-scale test bridge was constructed on the west campus of

Oklahoma State University. Figure 2-2 is a field photo of the test bridge (from west side of the bridge). All the real time results reported in this work were obtained from this bridge. To show the advantage of the heated bridge technology, only half of the test bridge is equipped with the heating system (right half of the bridge in Figure 2-2).

Figure 2-2: OSU Medium–Scale Smart Bridge

To better record the test results, a camera was installed on the test bridge (at northeast side of the bridge). It takes snapshots of the bridge every 15 minutes. Figure 2-3 shows a snapshot of the bridge. As mentioned before, to compare the effect of the heating system, only half of the bridge was equipped with the heating system. In the snapshot, the upper half of the bridge (the part inside the light rectangle) is where the heating system installed. An important note here, in the rest of this thesis, the bridge surface temperature refers to the surface temperature inside area equipped with the heating system!

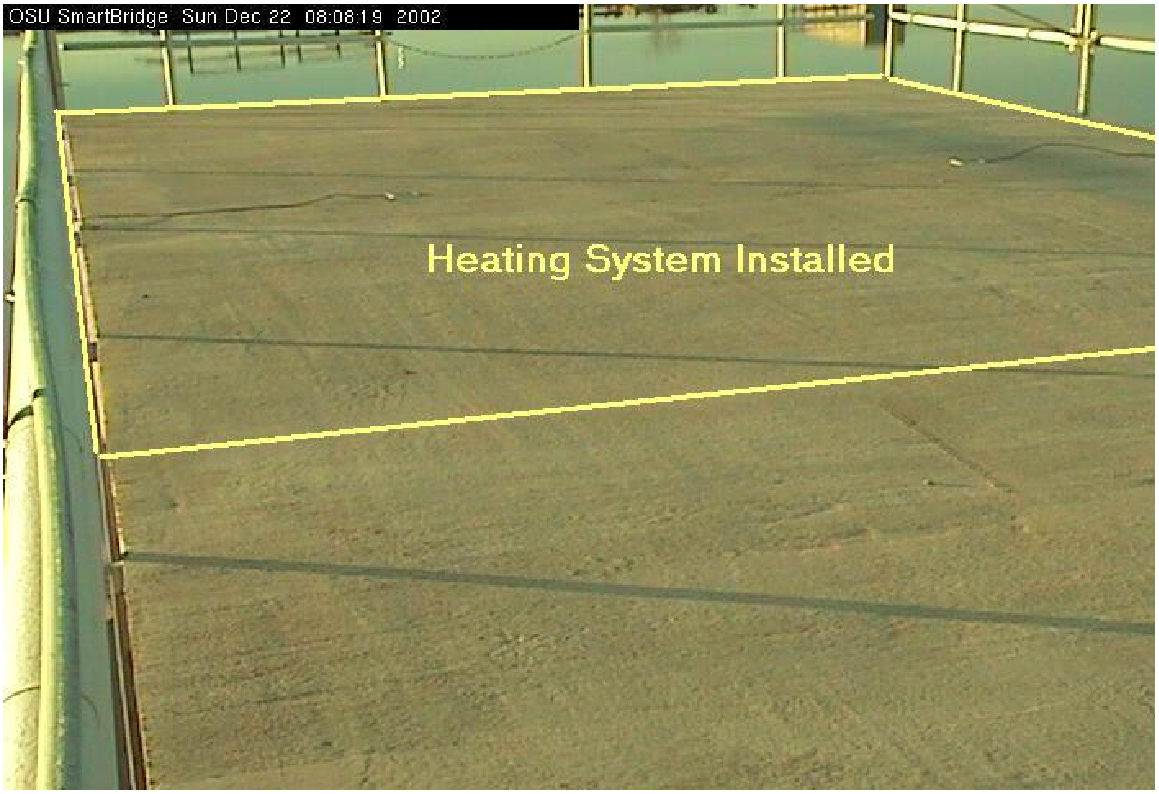


Figure 2-3: Bridge Snapshot

Figure 2-4 is a diagram of the OSU Geothermal Smart Bridge heating system. A is the pipe system installed under the bridge deck, called the Bridge Loop, which heats the bridge deck. E is the pipe system installed underground, called the Ground Loop, which gathers heat from ground. C is the heat pump, which transfers the heat from the Ground Loop to the Bridge Loop. F is the Smart Bridge Controller, which runs control strategy to turn on heat pumps when ice events are detected, changes the Bridge Loop supply flow temperature, and maintains bridge temperature. B and D are circulation pumps.

A full-scale installation will require multiple heat pumps (See Figure 2-1). The Bridge Loop supply flow temperature is changed by the numbers of heat pumps which are turned on. However, due to the scale of the OSU test bridge and the commercial available heat pump, only one heat pump was installed on the OSU Smart Bridge.

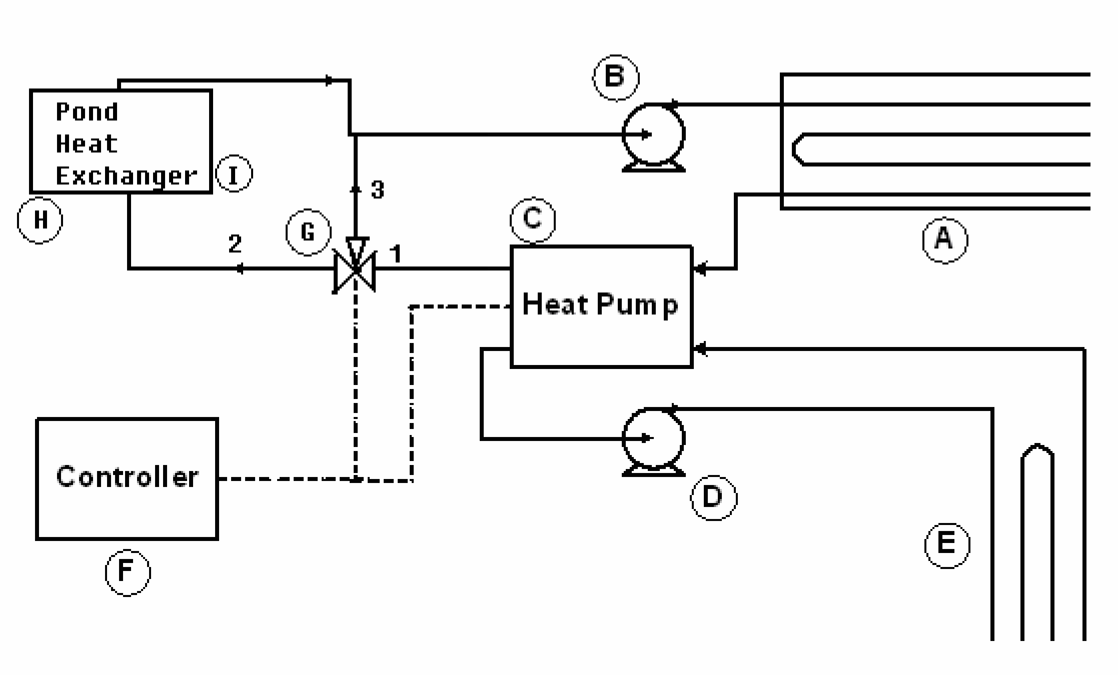


Figure 2-4: OSU Geothermal Smart Bridge Heating System

To simulate the condition of multiple heat pumps, an extra loop, called the Bypass Loop (I in Figure 2-4), is added to the Bridge Loop. Through the three way valve (G in Figure 2-4), the flow coming from the heat pump (stream 1) is divided into two streams, the bypass stream 2 and the main stream 3. Stream 2 is cooled by the exchanger H (stream 2). Then stream 2 (cool) and 3 (hot) are combined and go into the Bridge Loop. The temperature of the combined stream is determined by the ratio of stream 2 and 3, which is manipulated by the opening of the three way valve. Therefore, the Bridge Loop supply flow temperature is manipulated by the three way valve.

Through the three way valve and the Bypass Loop, the effect of multi heat pumps can be simulated by one heat pump.

2.6 Model Predictive Control

Model Predictive Control (MPC) is the name for a class of computer control schemes that utilize a process model for two central tasks: (1) explicit prediction of future plant behavior, (2) computation of appropriate corrective control actions required to drive the predicted output as close as possible to the desired target value [19]. In MPC, the current control action is obtained by solving a finite horizon optimal problem. Constraints can be included into the optimization. Due to its advantages over traditional feedback control, over the past twenty years, MPC has established remarkable industrial success. It is the most widely utilized advanced control algorithm in industrial applications.

Reference [18] gives a very good overview of the Model Predictive Control in the past, present and future. Several research hot spots can be found in [10].

Due to MPC's computational cost, most applications are in slow update systems, such as process industry, which usually has a time constant of minutes or hours. Recently, with the advance of computer hardware, researchers have begun to apply MPC in faster, nonlinear systems [7].

There are four basic elements in the MPC algorithm. They are: reference trajectory specification, process output prediction, control action sequence computation and error prediction update [19].

As mentioned in section 2.4, all existing heated bridge control systems use traditional feedback algorithm to control the heating system. Due to the nature of feedback, the heating system will be engaged only when those icing conditions have been detected. Since heating the bridge decks is a very slow procedure, usually hours, the bridge can not be heated in a short time. Snow or ice accumulation is inevitable at the beginning of the snow.

With the MPC algorithm, the bridge surface temperature prediction can be calculated. This gives the controller the ability to know what will happen in the future and act now. In other words, the bridge can be pre-heated from now if there is an icing condition detected in the future.

2.7 OSU Smart Bridge MPC Strategy

Different from existing Heated Bridge System control strategies, which are all traditional On/Off feedback controls with limited real time weather information, the OSU Geothermal Smart Bridge control system strategy utilizes advanced weather forecast to determine future ice events, then uses Model Predictive Control algorithm to heat the bridge by following a desired temperature trajectory. Therefore, it will pre-heat the bridge and keep the bridge deck temperature above freezing point prior to and through an expected icing event with minimum energy consumption while maintaining ice-free bridge conditions under any circumstances.

Previous research [13] has showed feasibility of the MPC algorithm in OSU Smart Bridge by several simulation case studies.

2.8 First Principle Bridge Deck Model

Section 2.7 mentioned that the OSU Smart Bridge controller uses a first-principle bridge deck model to predict the bridge deck surface temperature. This model was developed by investigators from the OSU Mechanical and Aerospace Engineering department. Complete information of the bridge deck model can be found in [4] and [21].

The bridge deck model uses a system of partial differential equations to describe the energy balance around a hydronically heated bridge deck. A two-dimensional finite difference approach is used to numerically solve this system of equations. The bridge deck model considers heat transfer due to solar radiation, thermal radiation, convection at the pavement surfaces, rain and snow evaporation, conduction through the bridge deck and tube walls, and heat transfer from the Bridge Loop fluid [4].

2.9 Rapid Update Forecast

As mentioned in 2.7, the OSU Geothermal Smart Bridge control system strategy utilizes advanced weather forecast to determine future ice events. The advanced weather forecast is obtained from the Rapid Update Forecast (RUC) system [1] [2].

The RUC system is a National Oceanic and Aeronautic Administration (NOAA) operational weather prediction system comprised primarily of a numerical forecast model and an analysis system to initialize that model. It was developed to serve users needing frequently updated short-range weather forecasts [1] [2].

The whole continental United States is divided in to grids. The RUC provides weather forecast for each grid. The first generation RUC system, which started running in 1994, provides 60-km resolution and 3-hour cycle. Then in 1998, the second generation

RUC system began to provide 40-km resolution and 1-hour cycle. The third generation RUC system was implemented in 2002, which provides 20-km resolution.

Every three hours, starting at 00:00 (GMT), RUC generates a 12-hour forecast. At the top of every hour (GMT) that is not a multiple of three, RUC outputs an updated 3-hour forecast. The RUC updates are shown in Figure 2-5.

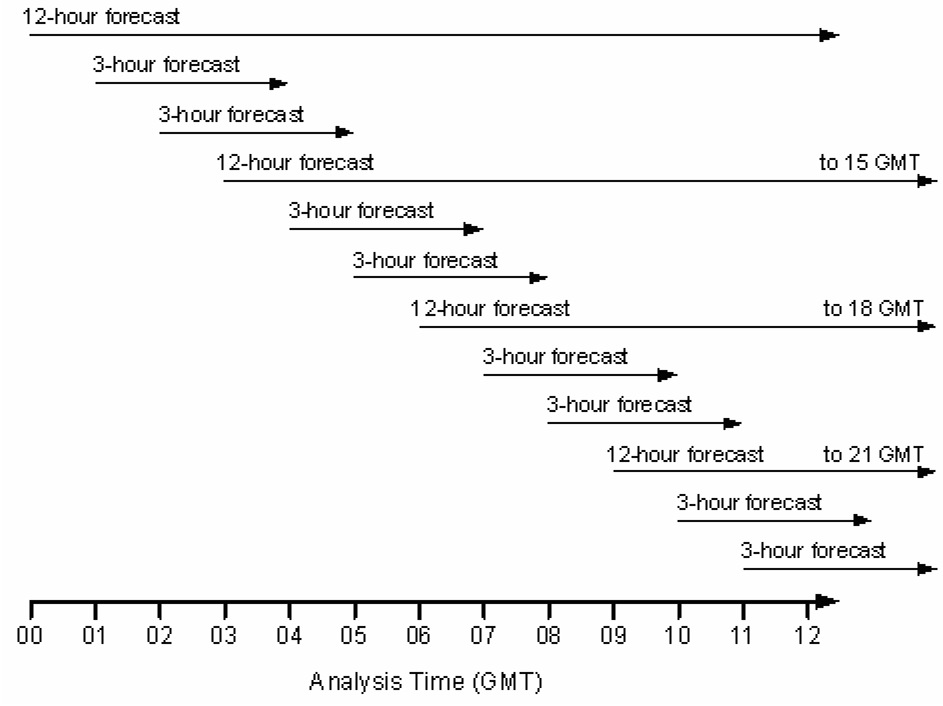


Figure 2-5: RUC Updates

The forecast weather information for each grid point is: temperature, relative humidity, wind direction, wind speed and rainfall (Table 2-2).

Because the weather information provided by the RUC system covers all of the continental United States, it is very easy to locate a heated bridge into a grid. Therefore, by using the RUC forecast, the Smart Bridge control system can be used at any point inside the U.S.

Temperature (°C)	Wind Speed (m/s)
Relative Humidity (%)	Rainfall (kg/m2)
Wind Direction (° from North)	

Table 2–3: RUC Forecast Weather Information

2.10 Oklahoma Mesonet

The continuous running of the Smart Bridge control system also needs bridge site weather information. This can be obtained by instruments installed on/near the bridge or from a nearby weather station.

The Oklahoma Mesonet [25] is a network of environmental monitoring stations. It consists of over 110 automated stations covering Oklahoma. There is at least one Mesonet station in each of Oklahoma's 77 counties.

The name "Mesonet" is a combination of the words "mesoscale" and "network". "Mesoscale" refers to weather events that range in size from a few kilometers to a few hundred kilometers [25].

Every five minutes, the observations at each site are transmitted to the Oklahoma Climatological Survey (OCS) at the University of Oklahoma (OU) and then become available to users.

The typical weather conditions monitored at a Mesonet station are listed in Table 2-3.

The nearest Mesonet station is only one mile from the OSU Smart Bridge. Therefore, Smart Bridge uses the Mesonet to obtain local weather information.

Figure 2-6 is a picture of the Mesonet station at Stillwater, OK.

Relative Humidity at 1.5 m (%)	Air Temperature at 1.5 m (°C)
Average Wind Speed at 10 m (m/s)	Vector Average Wind Speed (m/s)
Wind Direction at 10 m (° from N)	Standard Dev. of Wind Direction (° from N)
Standard Dev. Of Wind Speed (m/s)	Maximum Wind Speed (m/s)
Precipitation Since 00 GMT (mm)	Station Pressure (millibars)
Solar Radiation (W/m2)	Air Temperature at 9 m (°C)
Average Wind Speed at 2 m (m/s)	

Table 2–4 Oklahoma Mesonet Measurements

Figure 2-6: Stillwater Mesonet Station

CHAPTER III

SYSTEM IMPLEMENTATION

3.1 Control System Objective

The OSU Smart Bridge control system's objective is to develop an automatic control system which maintains ice-free bridge conditions while minimizing heated bridge life-cycle cost. The following specifications are integrated into the system design:

- Friendly user interface for operation and maintenance of the system by the end users.
- Modularity and flexibility to accommodate future changes and enhancements, including bridge model and control parameter adaptation.
- Robustness to the demands of a real-time operating environment.
- Capability to perform diagnostics, troubleshooting and performance evaluations.

3.2 System Hardware Architecture

Figure 3-1 is the current OSU Smart Bridge control system hardware architecture.

The two servers outside the rectangular area are located at the University of Oklahoma (OU). The RUC server receives the forecast weather information from the RUC system and relays it to the PetLab server. The Mesonet server stores all the observations from the Mesonet stations. Through these two servers, the Smart Bridge controller receives all the weather information needed.

All the components inside the rectangular area are located at Oklahoma State University (OSU). The PetLab Server and the PetLab PC are located in the Petroleum Lab (PetLab). The Bridge PC is installed at the bridge site. These three computers are connected by a local ethernet network.

The PetLab server is a Linux server. It connects to the OU servers by Wide Area Network (WAN) and serves as a Data Broker. There are two jobs running on this server. One is the RUC processing job. Every hour, a raw RUC data file, which contains all the forecast weather information for the U.S., is sent to the PetLab server. The RUC processing job will go through the file and generate a small .txt format file, which only contains the grid info near the Smart Bridge. Another job is Mesonet fetching job. Every five minutes, the job is triggered to fetch the observations from the Mesonet station near the Smart Bridge. The generated RUC files and the Mesonet files are stored on the PetLab server for future reference.

The Bridge PC runs the Slave Controller software. It has a Windows 98 system installed. Using add-in Analog to Digital (A/D) and Digital to Analog (D/A) boards, it has data acquisition and actuator abilities. It provides bridge site information, such as Bridge Surface temperature, Bridge Loop temperature, Bridge Loop flow rate, etc., to the PetLab PC and outputs control action from the PetLab PC to the heat pump and three way valve.

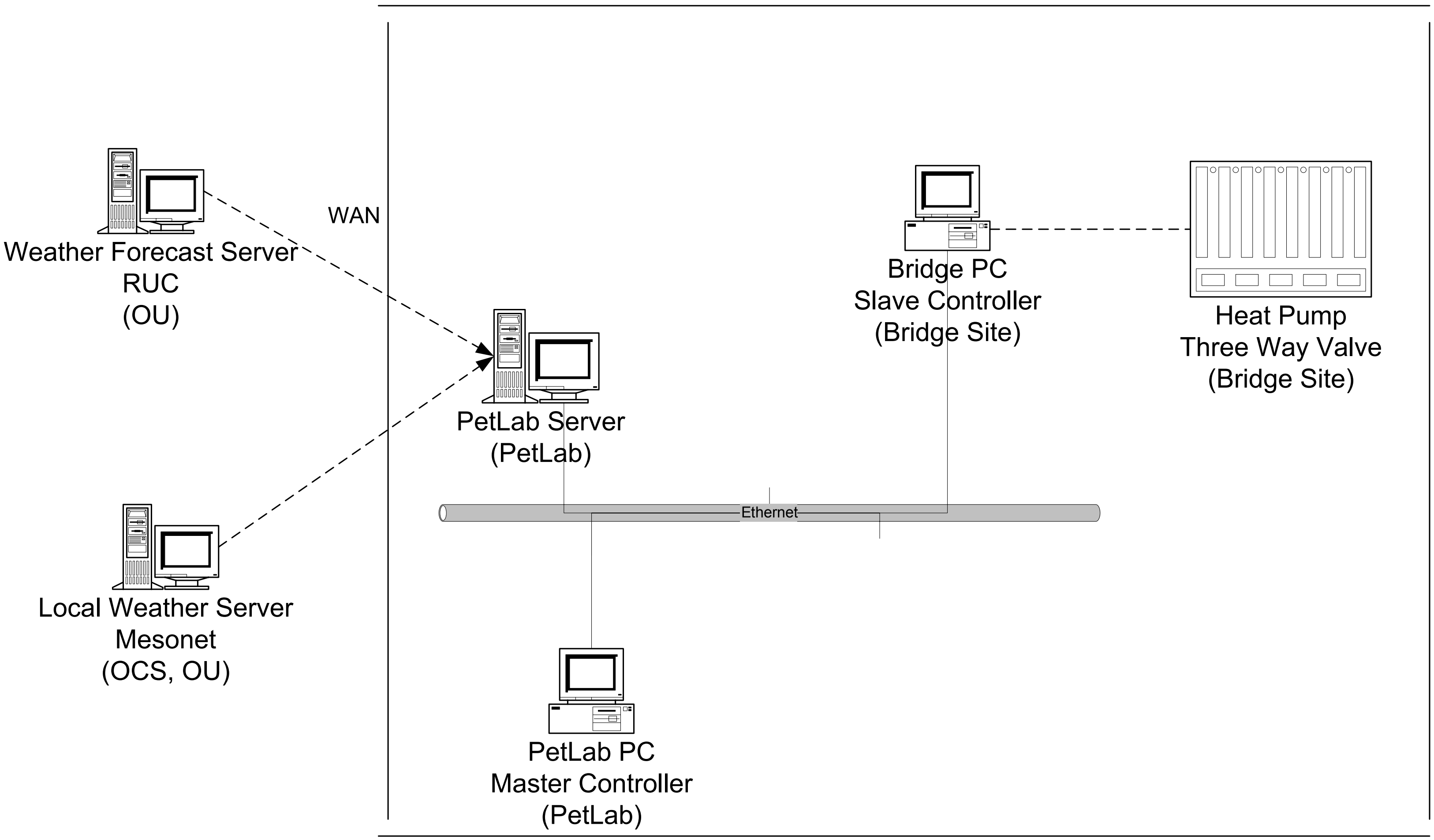


Figure 3-1: Control System Hardware Architecture

Master Controller software is running on the PetLab PC. It receives forecasted weather information to do the ice threat detection, runs the Model Predictive Control algorithm with the weather information and bridge site information, and generates control action to the Bridge PC. The PetLab PC runs a Windows 2000 operational system.

3.3 System Software Architecture

Figure 3-2 is the control system software architecture used in OSU Smart Bridge Project. It has a three-layer structure.

On the top is a rule-based meteorological feedforward controller. This controller checks the forecasted weather information and uses rules to identify whether there are icing events in the future. If the icing events exist, it will send an “On” signal to operate the heat pump and a desired bridge deck temperature response trajectory to the MPC controller.

The rules used in this controller are similar to the rules mentioned in Chapter II. However, because of the forecasted weather information, this controller can act much earlier than those. This makes the pre-heat procedure possible.

The MPC controller (Master Controller) stands at the middle of the framework. It uses the desired response trajectory, current bridge surface temperature and a first-principle bridge deck model to calculate the current control move – the Bridge Loop temperature, which acts as a set point to the Slave PID controller.

The meteorological feedforward controller is running on the PetLab Server and PetLab PC. The MPC controller is running on the PetLab PC.

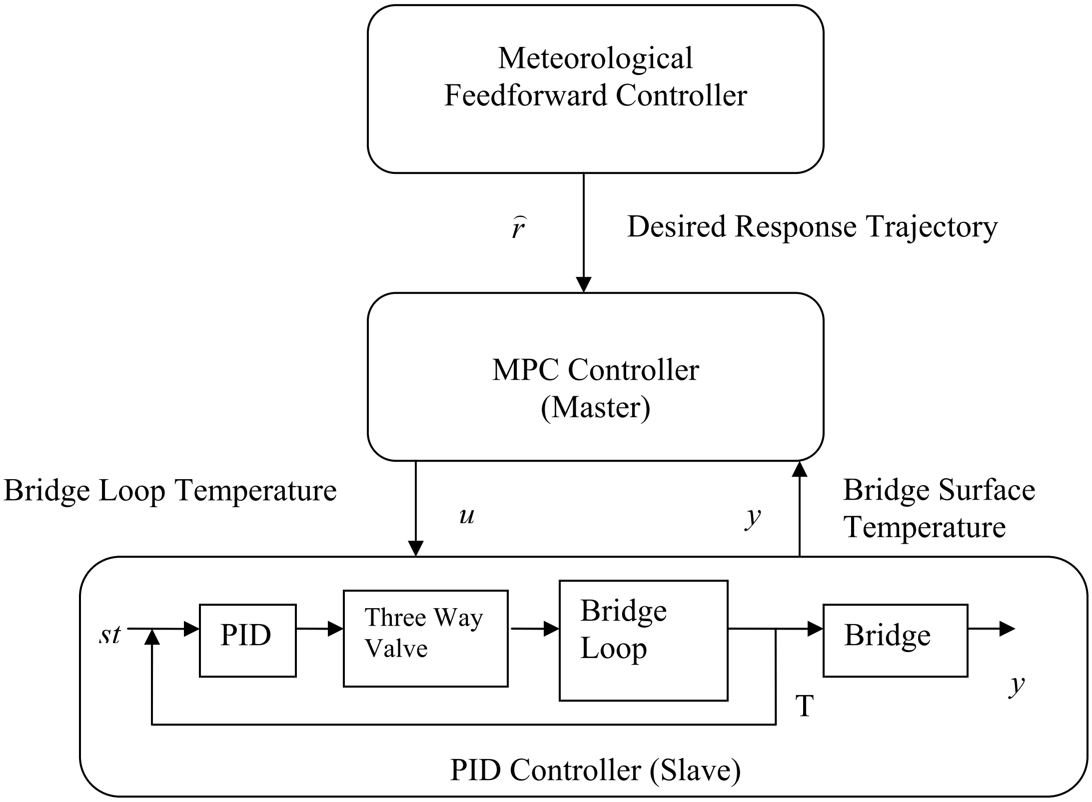


Figure 3-2: Control System Software Architecture

At the bottom of the framework is the PID controller (Slave Controller). The three way valve is manipulated by a conventional PID algorithm to control the Bridge Loop temperature. The MPC controller's output is the set point of the PID controller. Therefore, the MPC controller and the PID controller form a Master – Slave relationship. The PID controller is running on the Bridge PC.

The control system architecture is highly modularized. By careful design, only necessary information is transferred across the border of layers. The advantage is, by keeping the interface unchanged, each layer can be modified without affecting others. For example, currently the meteorological controller uses rules to determine icing threat. It can be modified to use Neural-Network to determine the icing threat.

3.4 Meteorological Controller

As described before, this meteorological controller checks the forecasted weather information, uses rules to determine icing events in the future, and generates desired bridge deck temperature response trajectory to the MPC controller. This section details each component of this controller.

3.4.1 Weather Forecast Processing Job

Every one hour, the OU RUC server sends the most recent generated RUC update to the PetLab server. However, this is a huge raw data file which contains all forecasted weather information of the continental U.S. The meteorological controller only needs the grid information where the Smart Bridge is located. This is done by a weather forecast processing program.

The weather forecast processing program is run on the PetLab server, written in Perl. Every ten minutes, this program is triggered by the operational system. This is through the Linux Cron Job mechanism.

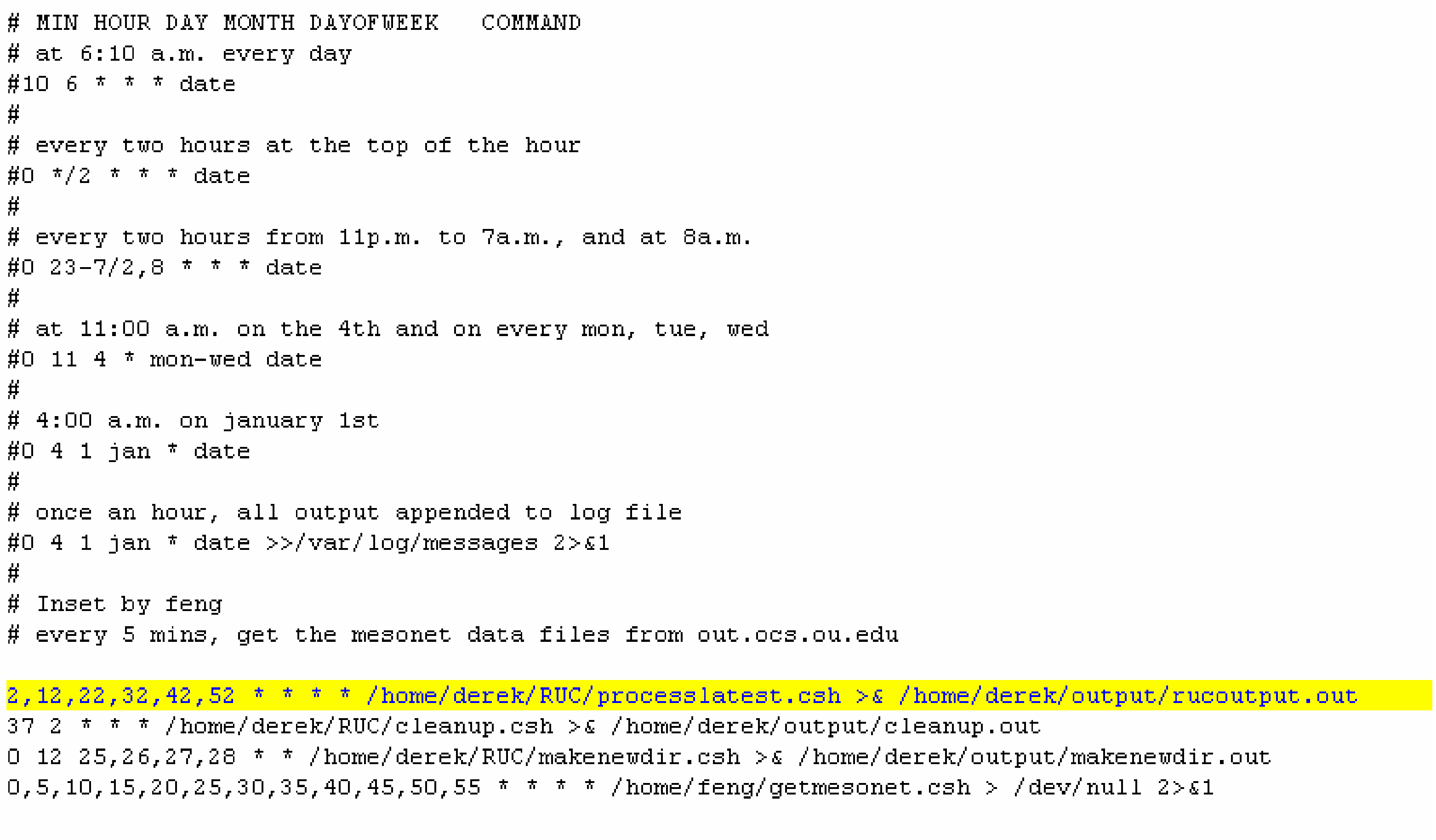
The Linux Cron Job is shown in Figure 3-3. The line with the shadow defines the time to trigger the processing program and where to put the processing results. All lines starting with symbol '#' are comments.

More detailed information about the RUC data file processing can be found at [6].

After the processing, the grid weather forecast information is recorded in a .txt format file. Figure 3-4 shows a typical 12-hour RUC forecast file. Forecasted weather information starts from the line with the shadow. This is a 12-hour forecast with three-hour intervals.

Each line contains nine columns. The first four columns are the record number, issue time, valid time and lead time. The number 200401111500 means 15:00 January 11, 2004. The time is based on Greenwich Mean Time (GMT).

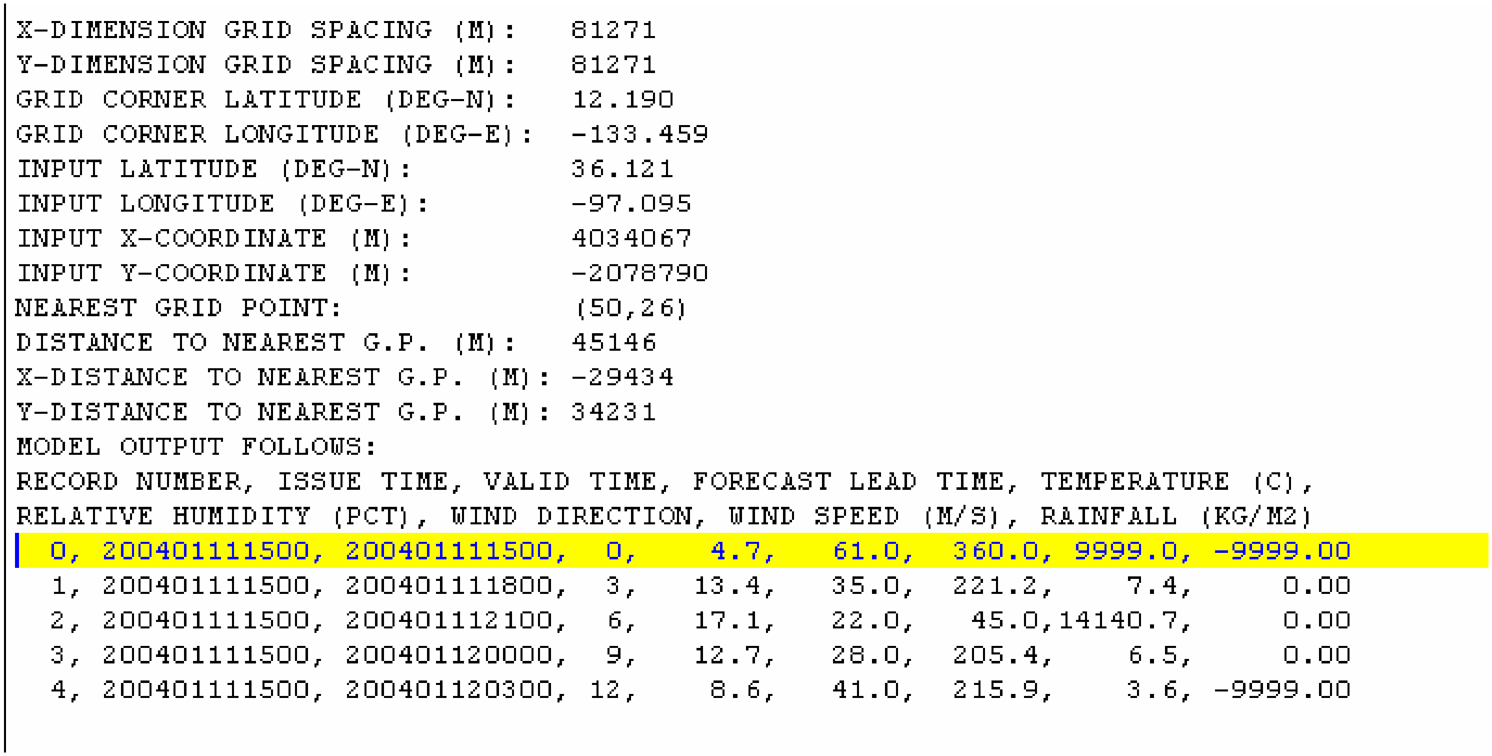
The rest of the five columns is the forecasted weather information summarized in Chapter II.



```
# MIN HOUR DAY MONTH DAYOFWEEK   COMMAND
# at 6:10 a.m. every day
#10 6 * * * date
#
# every two hours at the top of the hour
#0 */2 * * * date
#
# every two hours from 11p.m. to 7a.m., and at 8a.m.
#0 23-7/2,8 * * * date
#
# at 11:00 a.m. on the 4th and on every mon, tue, wed
#0 11 4 * mon-wed date
#
# 4:00 a.m. on january 1st
#0 4 1 jan * date
#
# once an hour, all output appended to log file
#0 4 1 jan * date >>/var/log/messages 2>&1
#
# Inset by feng
# every 5 mins, get the mesonet data files from out.ocs.ou.edu

2,12,22,32,42,52 * * * * /home/derek/RUC/processlatest.csh >& /home/derek/output/rucoutput.out
37 2 * * * /home/derek/RUC/cleanup.csh >& /home/derek/output/cleanup.out
0 12 25,26,27,28 * * /home/derek/RUC/makenewdir.csh >& /home/derek/output/makenewdir.out
0,5,10,15,20,25,30,35,40,45,50,55 * * * * /home/feng/getmesonet.csh > /dev/null 2>&1
```

Figure 3-3: Linux Cron Job



```
X-DIMENSION GRID SPACING (M):   81271
Y-DIMENSION GRID SPACING (M):   81271
GRID CORNER LATITUDE (DEG-N):   12.190
GRID CORNER LONGITUDE (DEG-E):  -133.459
INPUT LATITUDE (DEG-N):         36.121
INPUT LONGITUDE (DEG-E):        -97.095
INPUT X-COORDINATE (M):         4034067
INPUT Y-COORDINATE (M):         -2078790
NEAREST GRID POINT:             (50,26)
DISTANCE TO NEAREST G.P. (M):   45146
X-DISTANCE TO NEAREST G.P. (M): -29434
Y-DISTANCE TO NEAREST G.P. (M): 34231
MODEL OUTPUT FOLLOWS:
RECORD NUMBER, ISSUE TIME, VALID TIME, FORECAST LEAD TIME, TEMPERATURE (C),
RELATIVE HUMIDITY (PCT), WIND DIRECTION, WIND SPEED (M/S), RAINFALL (KG/M2)
  0, 200401111500, 200401111500,  0,    4.7,   61.0,  360.0, 9999.0, -9999.00
  1, 200401111500, 200401111800,  3,   13.4,   35.0,  221.2,    7.4,     0.00
  2, 200401111500, 200401112100,  6,   17.1,   22.0,   45.0,14140.7,     0.00
  3, 200401111500, 200401120000,  9,   12.7,   28.0,  205.4,    6.5,     0.00
  4, 200401111500, 200401120300, 12,    8.6,   41.0,  215.9,    3.6, -9999.00
```

Figure 3-4: 12-hour RUC Forecast File

3.4.2 Local Weather Fetching Job

The local weather conditions are obtained from the Oklahoma Mesonet.

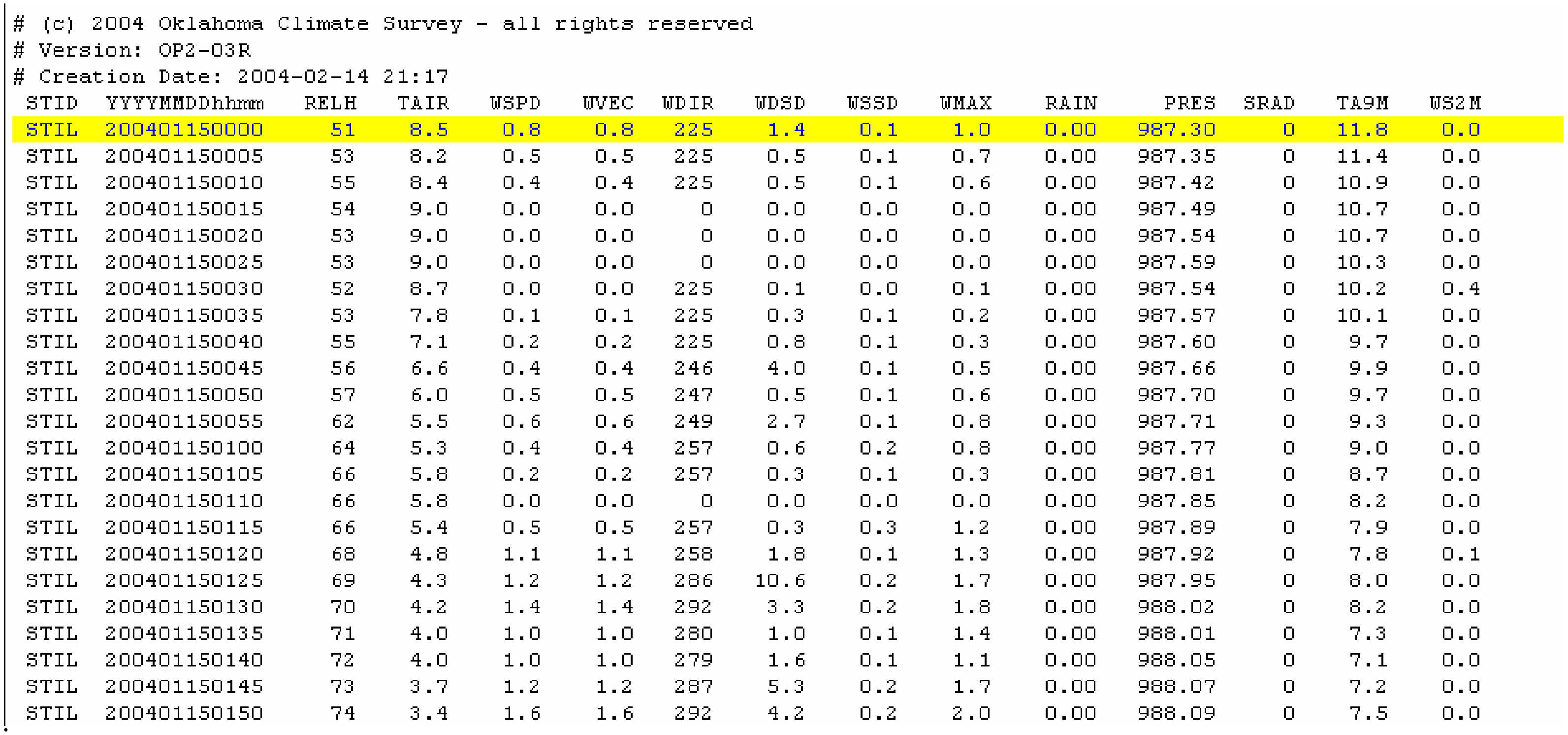
Everyday at 00:00 (GMT time), the OCS server will generate a .txt file for each Mesonet station in Oklahoma. During the following twenty four hours, this file will be continually updated by the weather information from the station.

Figure 3-5 shows part of a Mesonet file for the station in Stillwater, OK. Each weather record consists of a line that has fifteen columns. The first column is the station ID. The second column is the record time (in GMT). The remainders of the columns are the weather conditions.

To get Mesonet files, a local weather fetching script is run on the PetLab server. This script is triggered every five minutes by Linux Cron Job. The last line in Figure 3-6 shows this Cron Job. The script's name is "getmesonet.csh."

The "getmesonet.csh" script connects to the OCS server via ftp method. All the ftp commands are in the "netrc" script file. Figure 3-6 and 3-7 show these two script files.

With these scripts, the most recent Mesonet files are fetched and stored on the PetLab server.



```
# (c) 2004 Oklahoma Climate Survey - all rights reserved
# Version: OP2-03R
# Creation Date: 2004-02-14 21:17
 STID  YYYYMMDDhhmm  RELH   TAIR   WSPD   WVEC  WDIR   WDSD   WSSD   WMAX   RAIN     PRES  SRAD   TA9M   WS2M
 STIL  200401150000    51    8.5    0.8    0.8   225    1.4    0.1    1.0   0.00   987.30     0   11.8    0.0
 STIL  200401150005    53    8.2    0.5    0.5   225    0.5    0.1    0.7   0.00   987.35     0   11.4    0.0
 STIL  200401150010    55    8.4    0.4    0.4   225    0.5    0.1    0.6   0.00   987.42     0   10.9    0.0
 STIL  200401150015    54    9.0    0.0    0.0     0    0.0    0.0    0.0   0.00   987.49     0   10.7    0.0
 STIL  200401150020    53    9.0    0.0    0.0     0    0.0    0.0    0.0   0.00   987.54     0   10.7    0.0
 STIL  200401150025    53    9.0    0.0    0.0     0    0.0    0.0    0.0   0.00   987.59     0   10.3    0.0
 STIL  200401150030    52    8.7    0.0    0.0   225    0.1    0.0    0.1   0.00   987.54     0   10.2    0.4
 STIL  200401150035    53    7.8    0.1    0.1   225    0.3    0.1    0.2   0.00   987.57     0   10.1    0.0
 STIL  200401150040    55    7.1    0.2    0.2   225    0.8    0.1    0.3   0.00   987.60     0    9.7    0.0
 STIL  200401150045    56    6.6    0.4    0.4   246    4.0    0.1    0.5   0.00   987.66     0    9.9    0.0
 STIL  200401150050    57    6.0    0.5    0.5   247    0.5    0.1    0.6   0.00   987.70     0    9.7    0.0
 STIL  200401150055    62    5.5    0.6    0.6   249    2.7    0.1    0.8   0.00   987.71     0    9.3    0.0
 STIL  200401150100    64    5.3    0.4    0.4   257    0.6    0.2    0.8   0.00   987.77     0    9.0    0.0
 STIL  200401150105    66    5.8    0.2    0.2   257    0.3    0.1    0.3   0.00   987.81     0    8.7    0.0
 STIL  200401150110    66    5.8    0.0    0.0     0    0.0    0.0    0.0   0.00   987.85     0    8.2    0.0
 STIL  200401150115    66    5.4    0.5    0.5   257    0.3    0.3    1.2   0.00   987.89     0    7.9    0.0
 STIL  200401150120    68    4.8    1.1    1.1   258    1.8    0.1    1.3   0.00   987.92     0    7.8    0.1
 STIL  200401150125    69    4.3    1.2    1.2   286   10.6    0.2    1.7   0.00   987.95     0    8.0    0.0
 STIL  200401150130    70    4.2    1.4    1.4   292    3.3    0.2    1.8   0.00   988.02     0    8.2    0.0
 STIL  200401150135    71    4.0    1.0    1.0   280    1.0    0.1    1.4   0.00   988.01     0    7.3    0.0
 STIL  200401150140    72    4.0    1.0    1.0   279    1.6    0.1    1.1   0.00   988.05     0    7.1    0.0
 STIL  200401150145    73    3.7    1.2    1.2   287    5.3    0.2    1.7   0.00   988.07     0    7.2    0.0
 STIL  200401150150    74    3.4    1.6    1.6   292    4.2    0.2    2.0   0.00   988.09     0    7.5    0.0
```

Figure 3-5: Mesonet File

```
#! /bin/tcsh -f

ftp -i out.ocs.ou.edu
```

Figure 3-6: Getmesonet.csh

```
machine out.ocs.ou.edu
          login sbridge
          password *******

macdef init
          cd /usr/data/netshare/mesonet/text/t05
          ascii
          lcd /home/feng/mesonet
          mget 2004????stil.t05
          quit
```

Figure 3-7: Netrc

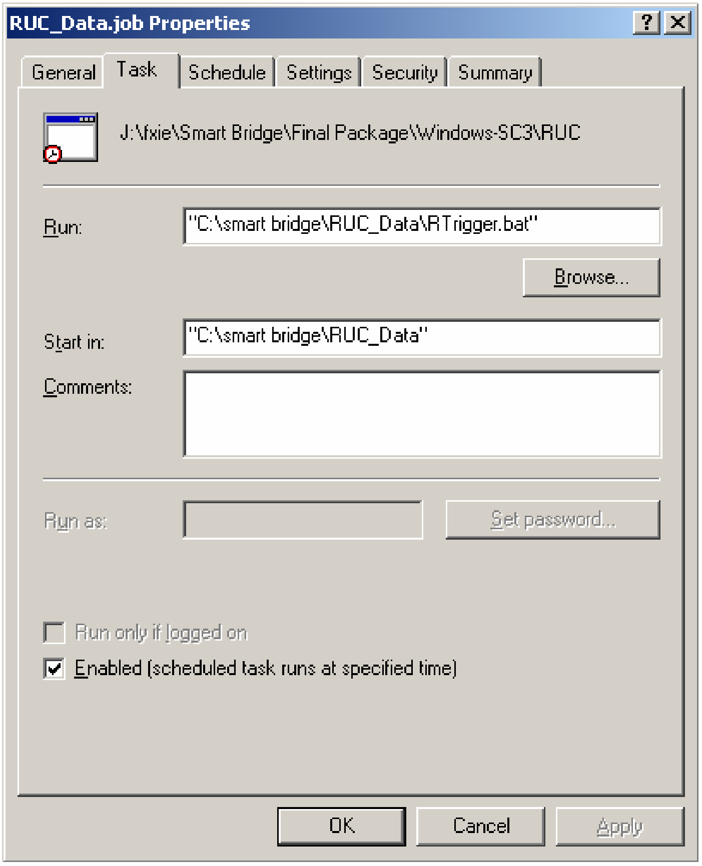
3.4.3 Weather File Transfer Job

The ice threat detecting and reference trajectory generating parts of the meteorological controller is run with the MPC controller on the PetLab PC. After RUC files and Mesonet files are generated and stored on the PetLab Server, these files need to be transferred to the PetLab PC. This job is done by Windows' Scheduler and FTP scripts running on the PetLab PC.

A RUC Data moving Task is setup in Windows' Scheduler (Figure 3-8). The trigger time is every ten minutes. Every time this "RTrigger.bat" task is awoken, a FTP script will be executed and will move all the RUC files from PetLab Server to PetLab PC. The script and the FTP commands are shown in Figure 3-9 and 3-10.

Similarly, Mesonet files will be moved every five minutes. The task is named “MTrigger.bat.” See Figure 3-11, 3-12 and 3-13 for this task and the FTP scripts.

The local ethernet network IP address of the PetLab Server is 192.168.0.1.



RUC_Data.job Properties

J:\fxie\Smart Bridge\Final Package\Windows-SC3\RUC

Run: "C:\smart bridge\RUC_Data\RTrigger.bat"

Start in: "C:\smart bridge\RUC_Data"

Comments:

Run as:

☐ Run only if logged on

☑ Enabled (scheduled task runs at specified time)

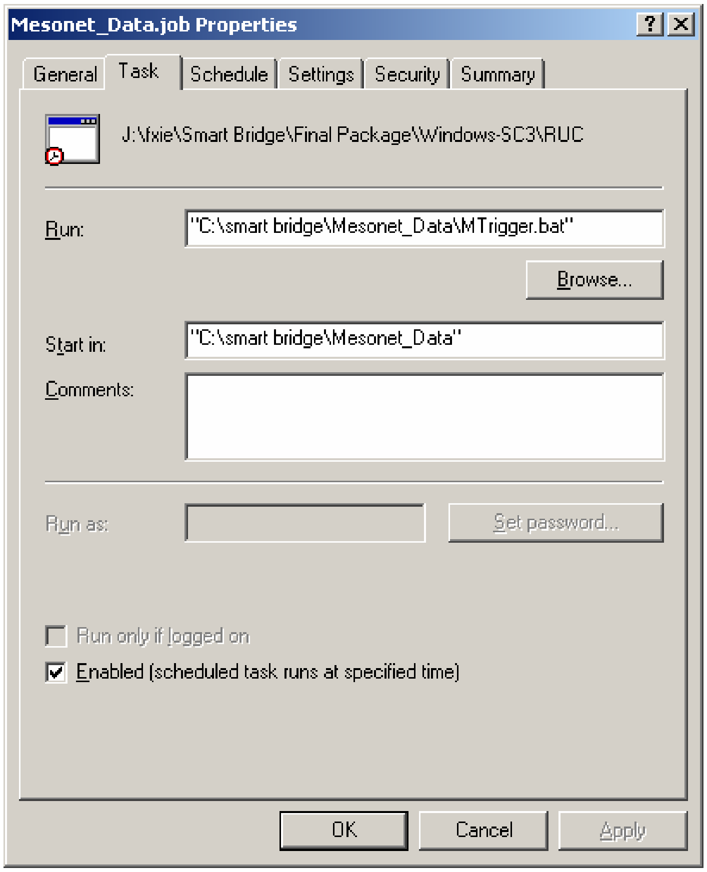
Figure 3-8: RUC Moving Task

```
ftp -i -s: RGetfile.txt 192.168.0.1
```

Figure 3-9: RTrigger.bat

```
feng
password
cd /home/derek/RUC/output/bridge01
ascii
mget 2004*
quit
```

Figure 3-10: RGetfile.txt



J:\fxie\Smart Bridge\Final Package\Windows-SC3\RUC

Run: "C:\smart bridge\Mesonet_Data\MTrigger.bat"

Start in: "C:\smart bridge\Mesonet_Data"

Comments:

Run as:

☐ Run only if logged on

☑ Enabled (scheduled task runs at specified time)

Figure 3-11: Mesonet Moving Task

```
ftp -i -s: MGetfile.txt 192.168.0.1
```

Figure 3-12: MTrigger.bat

```
feng
password
cd /home/feng/mesonet
ascii
mget 2004*
quit
```

Figure 3-13: MGetfile.txt

3.4.4 Weather Forecast Vector Generator

Though the RUC updates every hour, the forecast interval is three hours. Currently, the MPC controller’s sample time (User Adjustable) is 15 minutes. For a 12 hour forecast, the MPC controller needs a 48 point forecast vector.

The forecast vector generator works as follows (Figure 3-14):

Step One: From the current sample time, look back for the most recent 12-hour RUC update file and use this file to build the first vector, which has 5 points (the circles noted 0, 3, 6, 9, and 12 in Figure 3-14, Step One). The time interval between every adjacent two points is 3 hours. Then use linear interpolation to expand this vector into 13 points (the triangles in Figure 3-14, Step One are the interpolated ones). Now the time interval between every adjacent two points is 1 hour.

Step Two: Find the 3-hour RUC update files, which are issued after the previous 12-hour RUC update file and before the current sample time. Use the same linear interpolation, build a small vector with 4 points at 1 hour interval, and replace the vector points at the same time in the vector built in Step One.

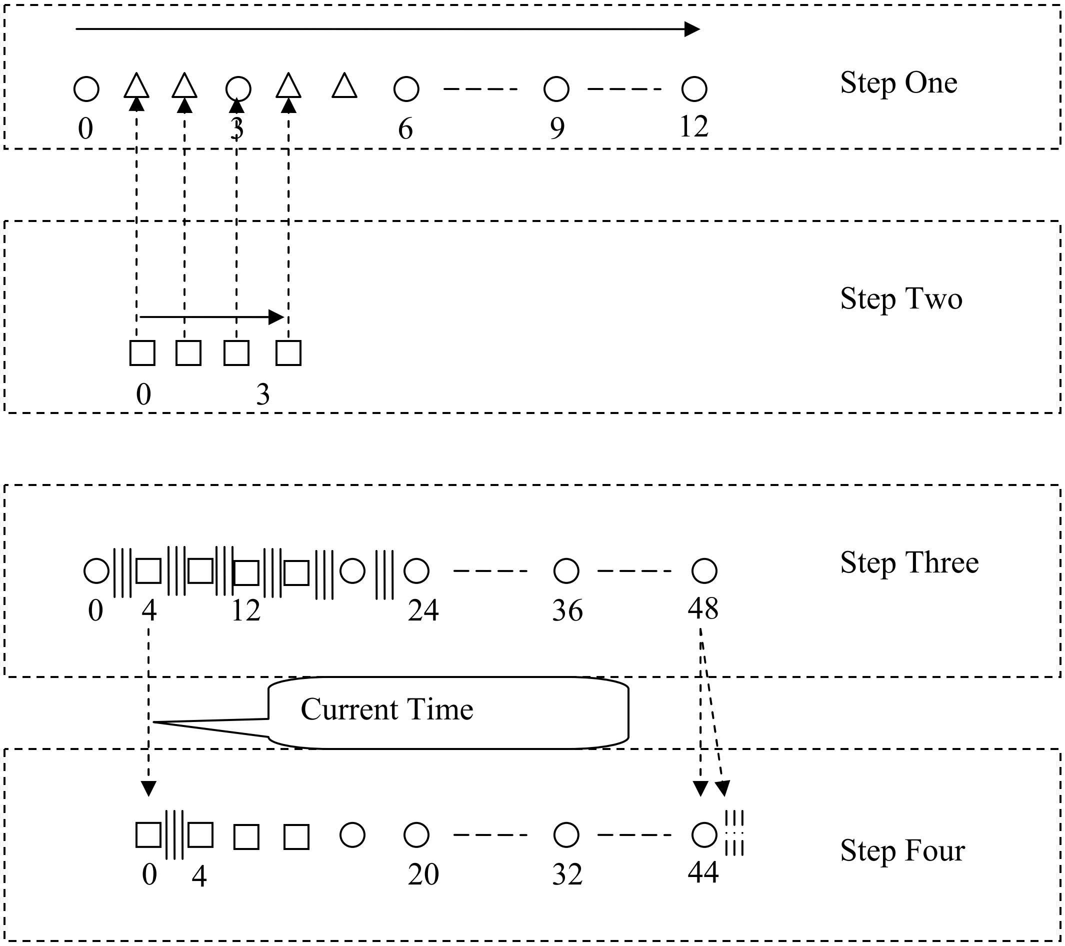


Figure 3-14: Weather Forecast Vector Generating

Step Three: Again, use linear interpolation to expand the vector built in Step Two into 49 points, with time intervals of 15 minutes (the vertical lines are the interpolated points).

Step Four: Start from current sample time, pick the points in the vector built in Step Three to form a new 48 point vector, and complete the end by extending the last point of the vector built in Step Three.

Each point in the weather forecast vector actually is a record, which contains six fields, time and the five weather data from the RUC update file (Refer to Table 2-2).

Two exceptions may happen during the weather forecast vector generation. One exception is file missing. Sometimes, one or more forecast files are lost. It is not so important if the missing file is a 3-hour forecast, because the 3-hour forecast is just a patch to the 12-hour forecast, without it, we still have the not very accurate 12-hour forecast. However, the situation becomes worse when some 12-hour forecast files are lost. The exception handling logic is like this, if the most recent 12-hour forecast file's issue time is already 12 hours before the current time, which means this forecast file has no weather information after the current time, use the current local weather information and the most recent 3-hour forecast file to build the first element of the forecast vector and use linear interpolation to expand the vector to the whole length of the vector.

The second exception is, sometimes, the content of the forecast file is meaningless. See Figure 3-4, there are some -9999.00 numbers in the file. This is due to some unknown reason during the transmission and data process. To handle this, data verification has been added to the forecast vector generator. When the data is not valid, the generator will search the same column in the previous record and use the most recent valid data to replace this bad data. For example, in case of forecast file of Figure 3-4, the predicted rainfall is -9999.00 in record four. The generator will search the same column in the third record and use 0.00 to replace this bad data. If the data in the third record is still invalid, then the generator will keep searching in the previous record until it finds a valid data.

3.4.5 Ice Threat Detection

After the weather forecast vector is built, the meteorological controller will go through this weather forecast vector and try to find the potential ice threat.

Ice threat detecting is rule-based. Two rules are used in the current controller. These rules are recommended by the investigators working on the OSU Smart Bridge Project Task 4.3.1.1: Weather Inputs.

The first rule is precipitation. A potential ice threat exists any time there is precipitation. This rule guarantees that the control system will attempt to drive the bridge deck surface temperature to the setpoint temperature at times when moisture is on the pavement. For cases when the average bridge surface temperature is far above the setpoint temperature, the control system will continue lowering the Bridge Loop supply temperature. When the Bridge Loop supply temperature reaches its lower constraint, the heating system will be set off. Therefore, the need to check the air temperature at the times of precipitation is not necessary.

The second rule is dew point depression. The dew point depression is calculated by Equation 3-1.

$$\text{Dew Point Depression} = \text{Air Temperature} - \text{Dew Point Temperature} \qquad \text{(Eq. 3-1)}$$

A potential ice threat exists any time the dew point depression falls below a specified threshold (User adjustable, currently is 2ºC). This rule guarantees that the average bridge deck surface temperature is above 0ºC any time the air has high moisture content.

Ice Threat Detection Rules

1. A potential ice threat exists any time there is precipitation.

2. A potential ice threat exists any time the dew point depression falls below a specified threshold

Table 3-1: Ice Threat Detection Rules

3.4.6 Reference Trajectory Generation

The reference trajectory is a desired average bridge deck surface temperature response when the potential ice threat exists. The trajectory is built in three steps.

First Step: By using the ice threat detection rules described in the previous section, the controller will scan the weather forecast vector and identify the first time an ice event will occur. From this point to the end of the forecast, the reference trajectory is set to setpoint above 0ºC. This set point is user adjustable. A typical value is 2ºC

Second Step: To provide a margin of safety, the reference trajectory is extended a certain hours before the first ice threat. This is also a user adjustable variable. A typical value is 1.5 hours.

Third Step: Now the only part left of the reference trajectory is from the current time to the safety margin. This part is set to a temperature ramp. This will make the whole reference trajectory like a first order system response. The slope of the temperature ramp is a user specified parameter. This parameter should be chosen very carefully. An improper value will result in an unfeasible trajectory that the bridge deck heating system will not be able to supply enough heat or will result in a unnecessary long operation time.

Figure 3-15 shows the steps to build the reference trajectory.

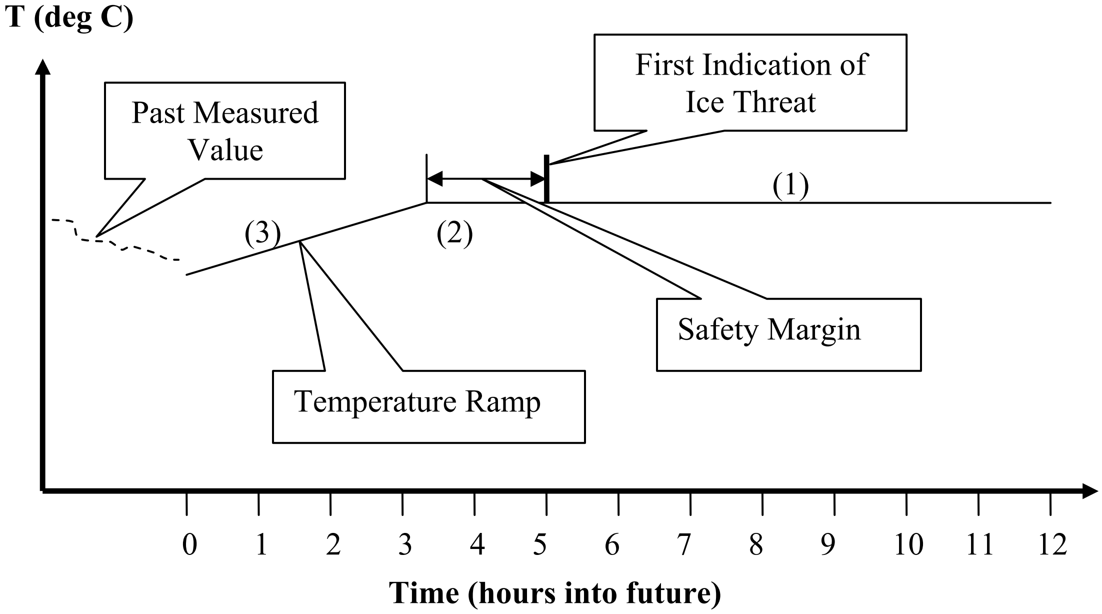


Figure 3-15: Reference Trajectory Building Steps

3.5 MPC Controller (Master)

After the meteorological detected the incoming ice threat, the reference trajectory is updated with the MPC execution frequency (user adjustable, 1/sample time) until no potential ice threat exists. The Model Predictive Controller calculates the control action to drive the bridge deck surface temperature to the reference trajectory as close as possible. The MPC algorithm used is a Quadratic Dynamic Matrix Control (QDMC).

As described in Chapter II, the bridge heating system is a Single Input Single Output (SISO) system. The manipulate variable (MV) is the Bridge Loop supply temperature. The controlled variable (CV) is the bridge deck surface temperature.

The QDMC algorithm used in the OSU Smart Bridge control system can be described as follows:

At each sample time, the Smart Bridge controller uses a first-principle bridge deck model to predict the bridge deck surface temperature response with forecasted weather information and the assumption of no future control actions (Figure 3-16). The predicted bridge deck response is noted as $\hat{y}$.

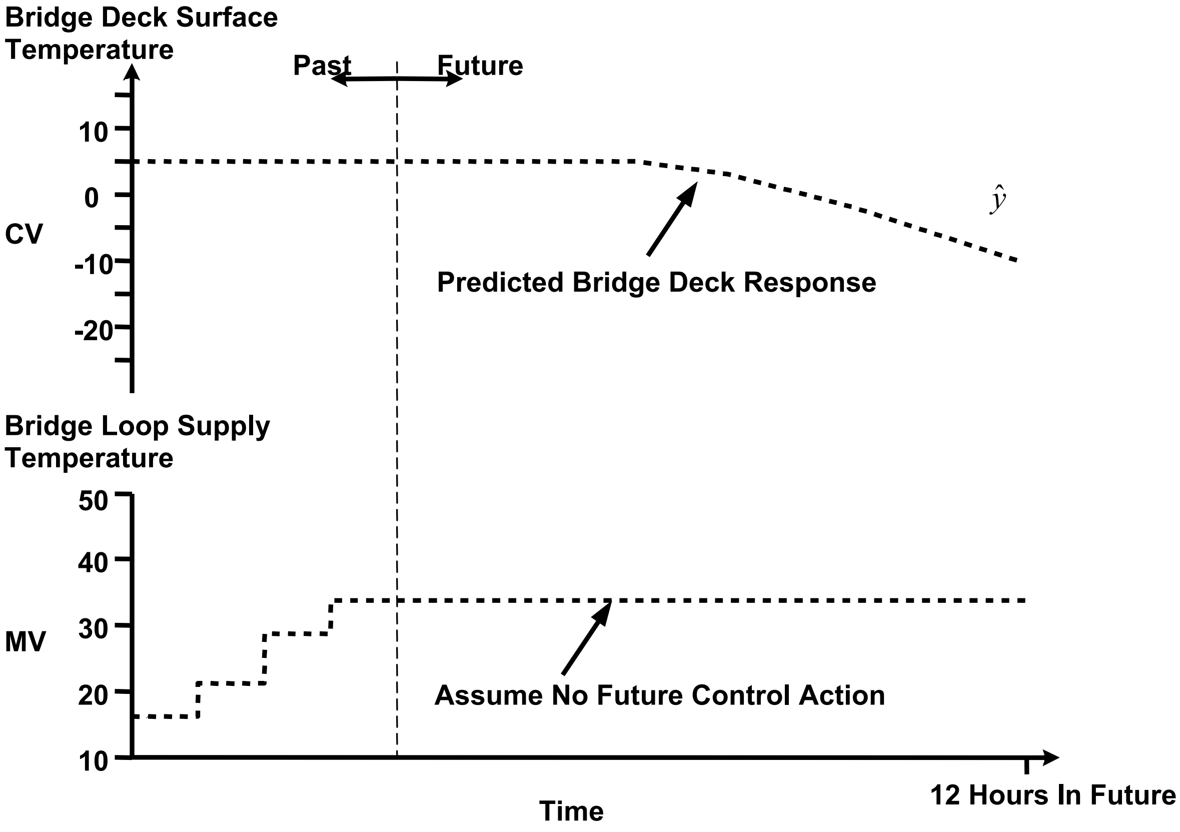


Figure 3-16: MPC Algorithm Step One

The difference (projected error vector) between the predicted bridge deck response and the desired response trajectory is calculated $\hat{e} = \hat{r} - \hat{y}$ (Figure 3-17).

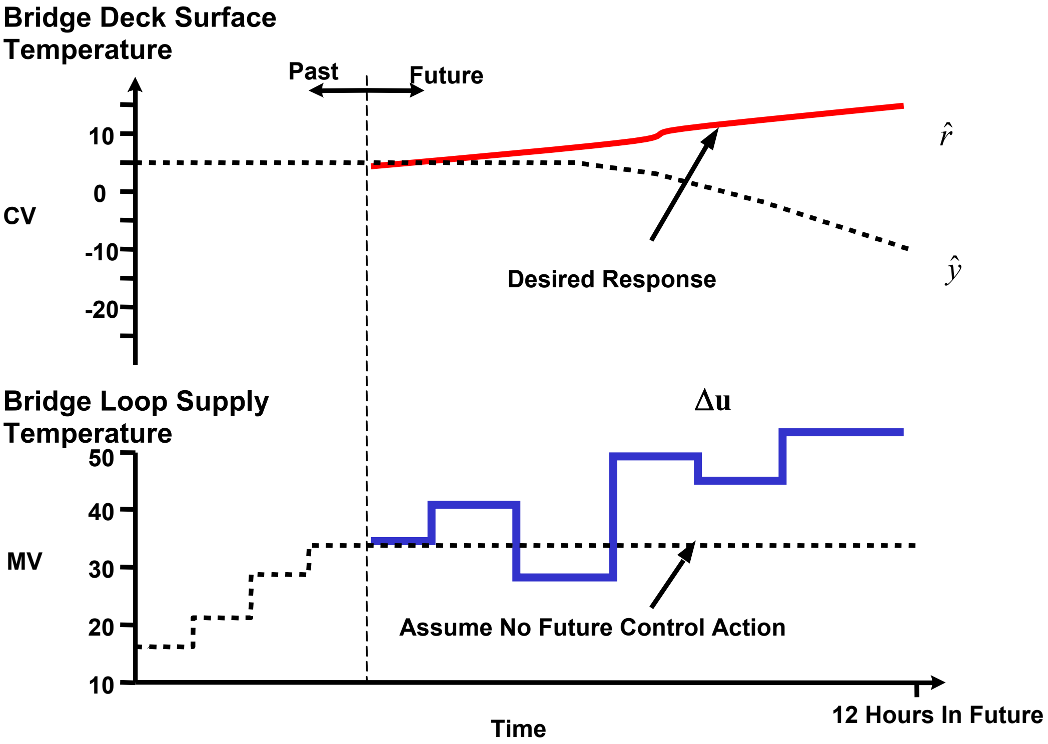


Figure 3-17: MPC Algorithm Step Two

Then by solving the following optimization problem (equation 3-2 and 3-3), the future MV (Bridge Loop supply temperature) adjustments (**Δu**) is calculated. This is a quadratic problem with linear constraints.

After the **Δu** vector is obtained, the first element of this vector is added to the previous control action *u* to form the current control action. This adjusted control action is sent to the process.

At the next sample time, all the steps are repeated.

$$\min_{\Delta u} \Phi = (\hat{e} - A\,\Delta u)\Gamma^T\Gamma(\hat{e} - A\,\Delta u) + \Delta u^T \Lambda^T \Lambda\,\Delta u \qquad \text{(Eq. 3-2)}$$

$$\text{s.t.} \qquad u_{min} \le u \le u_{max} \qquad \text{(Eq. 3-3)}$$

where: $\boldsymbol{\Phi}$ = objective function

$\hat{e}$ = projected error vector = $\hat{r} - \hat{y}$

$\mathbf{A}$ = dynamic matrix

$\mathbf{\Delta u}$ = sequence of future MV adjustments

$\mathbf{\Gamma}$ = output error weighting matrix

$\mathbf{\Lambda}$ = input error weighting matrix

The first term on the right hand side of Equation 3-2 is called the error penalty term. The purpose of this error penalty term is to penalize the discrepancies between the predicted response ($\hat{y}$) without control action and the desired response ($\hat{r}$). The second term on the right hand side of Equation 3-2 is called the move suppression term. The purpose of the move suppression term is to penalize the large moves in the manipulated variable.

Equation 3-3 defines the constraints imposed on the MV. The Bridge Loop supply temperature is constrained into a certain range of values by the limits of the heat pump.

Qualitatively, the optimization problem given in Equations 3-2 and 3-3 is to find a sequence of MV adjustments (**Δu**) that minimize the discrepancies between $\hat{y}$ and $\hat{r}$, but not at the expense of making unacceptably large MV adjustments. Also these MV adjustments must not violate the constraints.

Matrix **A** is called the dynamic matrix. It is defined in Equation 3-4. It has *p* rows and *m* columns. The *p* is called the prediction horizon and the *m* is called the control horizon.

$$\mathrm{A} = \begin{bmatrix} a(1) & 0 & 0 & \ldots & 0 \\ a(2) & a(1) & 0 & \ldots & 0 \\ a(3) & a(2) & a(1) & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a(m) & a(m-1) & a(m-2) & \ldots & a(1) \\ a(m+1) & a(m) & a(m-1) & \ldots & a(2) \\ a(p) & a(p-1) & a(p-2) & \ldots & a(p-m+1) \end{bmatrix} \qquad \text{(Eq. 3-4)}$$

Since the weather forecasting product (RUC update) used in this control system provides a 12-hour forecast, a 12-hour prediction horizon is chosen. Using a 15-minute sample time, *p* equals 48.

As for the control horizon, a typical choice is one fourth to one third of the prediction horizon. The current controller uses an *m/p* ratio of one half. Therefore, the control horizon is 6 hours. At the 15-minute sample time condition, *m* equals 24.

The elements (*a*(k)) in the Dynamic Matrix **A** come from the unit step response model of the process. In this system, the manipulate value (MV) is the Bridge Loop supply temperature and the controlled value (CV) is the bridge deck surface temperature. The unit step response reflects the change of the bridge deck surface temperature to a

+1°C step change in the Bridge Loop supply temperature under constant weather conditions. A vector, **a,** is used to represent this MV-CV step response model.

The unit step response is carried off-line before the controller runs. After the vector **a** obtained, the matrix **A** is fixed when the controller is running.

Figure 3-18 shows the unit step response. *a*(k) are recorded at each sample time (The same as the controller sample time, current is 15 minutes) and with these values, the Dynamic Matrix **A** is constructed in the manner of Equation 3-4.

The value shown in Figure 3-18 is in deviation.

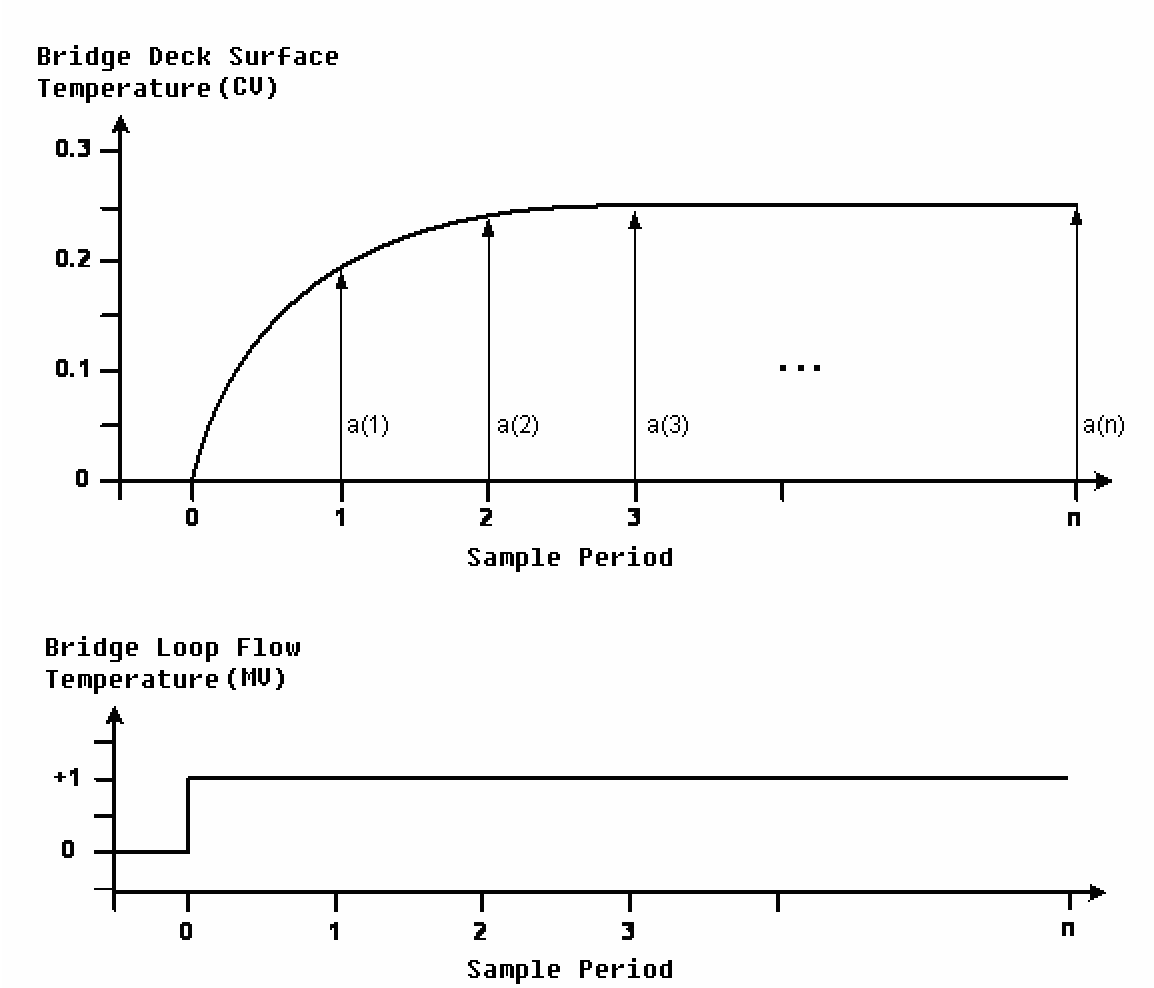


Figure 3-18: Unit Step Response

The weather conditions used to produce step response are listed in Table 3-2.

Variable	**Value**
Heating Fluid Flowrate (kg/s)	2
Heating Fluid Temperature (degC)	30
Ambient Air Temperature (degC)	-13
Relative Humidity (%)	81
Wind Speed (m/s)	0
Wind Direction (deg from North)	0
Solar Radiation (W/m^2)	0
Soar angle of incidence (Rad)	0.785
Snowfall in water equivalent (mm/hr)	6.35
Rainfall in water equivalent (mm/hr)	6.35

Table 3-2: Weather Conditions Used to Produce Step Response

The effect of the MV adjustments is calculated by the term $\mathbf{A\Delta u}$. The term $\hat{e}$ - $\mathbf{A\Delta u}$ is called the residual error. The residual error will be minimized when a sequence of MV adjustments is selected, which lets $\mathbf{A\Delta u}$ compensate $\hat{e}$. Therefore, the error penalty term in Equation 3-2 will be minimized.

The Γ matrix is called the output error weighting matrix. It is a diagonal (p x p) matrix. It is defined in Equation 3-5. The elements on the diagonal are noted as vector γ.

$$\mathbf{\Gamma} = \begin{bmatrix} \gamma(1) & 0 & ... & 0 & 0 \\ 0 & \gamma(2) & ... & 0 & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & ... & \gamma(p-1) & 0 \\ 0 & 0 & ... & 0 & \gamma(p) \end{bmatrix} \quad \text{(Eq. 3-5)}$$

The elements of the vector γ serve as tuning parameter to the objective function. It will affect the aggressiveness of the controller. A previous study suggested the parameter as $\gamma(1) \sim \gamma(16) = 2.5$, $\gamma(16) \sim \gamma(32) = 1.0$, and $\gamma(33) \sim \gamma(48) = 0.7$ [13].

The **Λ** matrix is called the input weighting matrix. It is a diagonal (*m* x *m*) matrix. See its definition in Equation 3-6. The elements on the diagonal are noted as vector λ.

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda(1) & 0 & ... & 0 & 0 \\ 0 & \lambda(2) & ... & 0 & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & ... & \lambda(m-1) & 0 \\ 0 & 0 & ... & 0 & \lambda(m) \end{bmatrix} \quad \text{(Eq. 3-6)}$$

Elements of λ give weight on each elements of the sequence of MV adjustments (**Δu).** The smaller of the λ, the larger of the **Δu. A** previous study suggested the parameter as $\lambda(1) \sim \lambda(24) = 1$ [13].

3.5.1 Bridge Deck Surface Temperature Prediction

The bridge deck surface temperature prediction $\hat{\mathbf{y}}$ is calculated by a first-principle bridge deck model. This bridge deck model can be summarized in Figure 3-19. With all the input parameters and variables, the bridge deck model can predict the outputs at time *t* in the future. Currently, the MPC controller's prediction horizon is 48 points (based on the 12 hours weather forecast time and the current controller sample time of 15 minutes). Therefore, bridge deck surface temperature prediction $\hat{\mathbf{y}}$ is also 48 points. The parameter *t* is 15 minutes in this case.

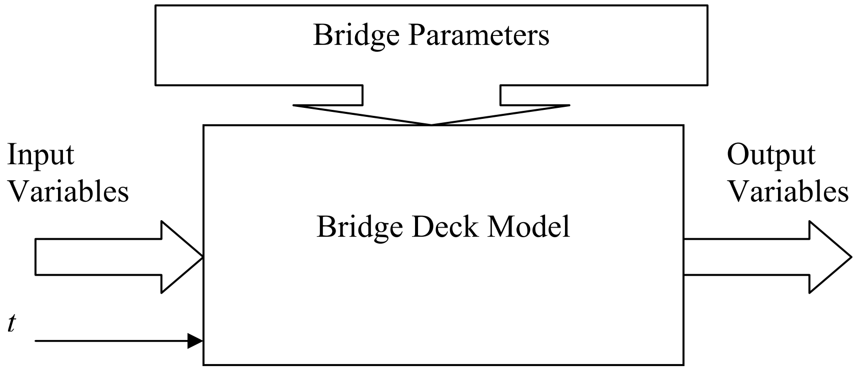


Figure 3-19: First-Principle Bridge Deck Model

At each controller execution instance, the bridge deck model is called 48 times to generate the bridge deck surface temperature prediction $\hat{\mathbf{y}}$.

The bridge parameters are the physical specification of the bridge. They must be specified before Smart Bridge control system is activated. Table 3-3 lists all the parameters required by the bridge deck model.

Each time the bridge deck model is used, a set of input variables and a time *t* (minutes) are passed to the model. Then the model will calculate and predict the bridge conditions *t* minutes in the future under current input variables. Table 3-4 lists all the input variables. Most of the input variables are from the RUC Weather Forecast and bridge site measurements. Others, such as Sky Temperature and Solar Angle of Incidence are from calculations. Details can be found by contacting the investigators working on the OSU Smart Bridge Project Task 4.3.1.1: Weather Inputs.

Pavement Length (m)	Absorptivity Coefficient (dimensionless)
Pavement Width (m)	$C_{p,Layer\ 1}$ (J/m^3 °C)
Slab Orientation (° from North)	$C_{p,\ Layer\ 2}$ (J/m^3 °C)
Pavement Thickness (m)	k_{Pipe} (W/m °C)
Pipe Spacing (m)	Wall Thickness of Pipe (m)
Pipe Diameter (m)	Fluid Type (2 for GS-4)
Pipe Depth Below Surface (m)	Weight % GS-4 (%)
Depth to Interface 1 (m)	Number of Flow Circuits
$k_{Layer\ 1}$ (W/m °C)	Length of Pipe Per Circuit (m)
$k_{Layer\ 2}$ (W/m °C)	Transient Time Step (sec)
Emmissivity Coefficient (0.9)	Bottom Boundary Condition
Minimum Flow Condition (kg/sec)	

Table 3-3: Bridge Parameters

The output variables are listed in Table 3-5. Those are the predicted bridge conditions.

Using this bridge deck model and the weather forecast, a 12-hour bridge surface temperature prediction can be calculated.

After prediction $\hat{\mathbf{y}}$ is calculated, the current bridge surface temperature measurement is used to adjust the prediction.

Air Temperature (°C)	Solar Radiation (W/ m^2 °C)
Humidity Ratio (kg water/kg dry air)	Solar Angle of Incidence (radians)
Sky Temperature (°C)	Snowfall Rate (mm/hr water equivalent)
Wind Speed (m/sec)	Rainfall Rate (mm/hr water equivalent)
Wind Direction (° from North)	Bridge Loop supply temperature (°C)
Bridge Loop flowrate (kg/sec)	

Table 3-4: Input Variables

Average Bridge Deck Surface Temperature (°C)
Bridge Loop Return Temperature (°C)
Heat Transfer Rate From Bridge Loop (kJ/sec)

Table 3-5: Output Variables

3.5.2 Optimization Technique

The optimization problem given in Equation 3-2 and 3-3 is solved by a cyclic method with line search [11].

A cyclic method is an iterative technique used to minimize multiple variable objective functions. In each iteration, all the variables are optimized one by one. For example, for a two variable objective function, $\min \Phi = f(x, y)$, each iteration is carried out like this: first, y is fixed and the objective function is minimized with x, then x is fixed and the objective function is minimized with y. These two steps combine one iteration of the cyclic method.

Because the MV adjustment (**$\Delta\mathbf{u}$)** in Equation 3-2 is a vector with *m* elements (currently it is 24), each iteration in the cyclic search consists of *m* individual searches. The cyclic method is summarized as follows:

1. Assign the initial guess $\mathbf{\Delta u^0}$**,** iteration counter = 0
2. Evaluate objective function at $\Phi(0)$ at $\mathbf{\Delta u^0}$
3. Use the line search method, optimize each element of $\mathbf{\Delta u^0}$ one by one (*m*) and form the new $\mathbf{\Delta u^1}$
4. Evaluate objective function $\Phi(1)$ at $\mathbf{\Delta u^1}$
5. Test stopping criteria, if satisfied, go to (7)
6. Assign $\mathbf{\Delta u^1}$ to $\Delta\ \mathbf{u^0}$, assign $\Phi(1)$ to $\Phi(0)$ increase iteration counter by 1, go back to step (3)
7. Found the optimized $\mathbf{\Delta u^* = \Delta u^1}$**,** Stop

The stopping criteria for the cyclic method are listed as follows. The algorithm stops when any of these criteria are satisfied.

- If $|\Phi(0) - \Phi(1)| < 1$
- If the iteration counter reaches 12
- If all $|\Delta u_i^0 - \Delta u_i^1| < 1$, i=1,2,3…24, Δu_i^0 and Δu_i^1 means the ith element of the vector $\mathbf{\Delta u^0}$ and $\mathbf{\Delta u^1}$

A two-point equal interval region elimination algorithm is used in the line search of the cyclic method described above.

The line search method is summarized as follows:

(1) For a certain Δu_i, a search boundary [a, b] is assigned, Δu_i means the the ith element of the vector $\mathbf{\Delta u}$

(2) Calculate $\rho_1 = a + (b - a)/3$, $\rho_2 = b - (b - a)/3$

(3) Evaluate objective function at $\Delta u_i = a$, $\Delta u_i = b$, $\Delta u_i = \rho_1$ and $\Delta u_i = \rho_2$

(4) Eliminate region with highest objective function value

(5) Test stop criteria, if satisfied, go to (7)

(6) Reassign a and b, go to (2)

(7) Select the optimized Δu_i^* among a, b, ρ_1 and ρ_2 with minimum objective function value

This algorithm is illustrated in Figure 3-20.

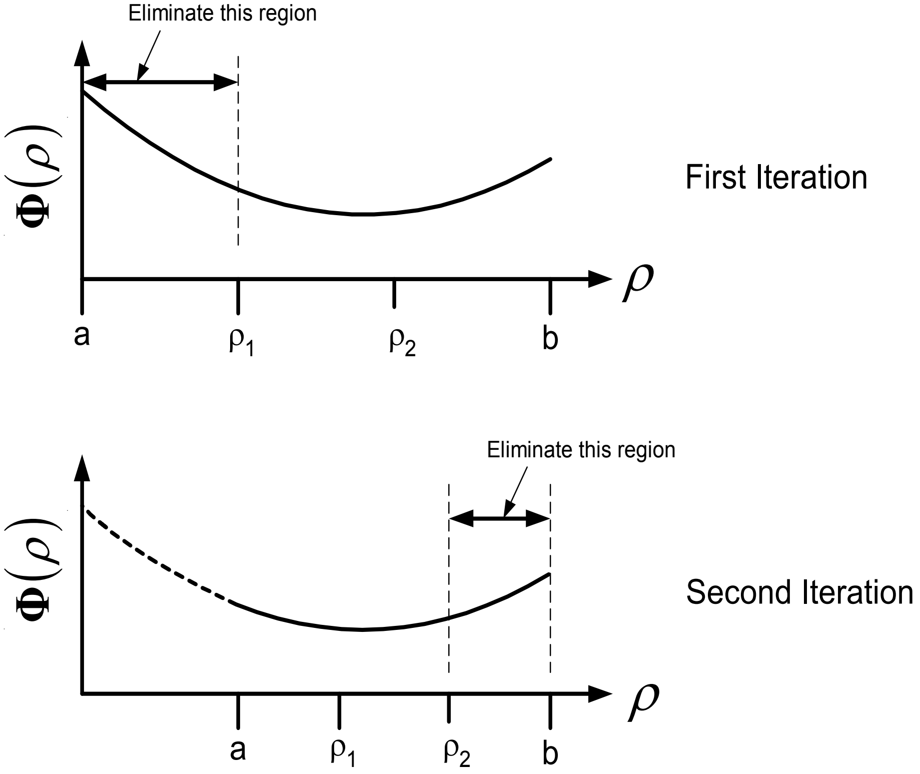


Figure 3-20: Two-point Equal Interval Line Search

After the first iteration, ρ_1 becomes a. After the second iteration, ρ_2 becomes b.

The stop criterion is $|\rho_1 - \rho_2| < 0.1$.

The cyclic and the two-point equal interval region elimination algorithms are very direct. They are easy to understand and program. Thus the codes are easy to maintain.

Both of the algorithms need lots of calculation. Compare to other algorithms, such as the gradient methods, these two are not efficient. However, because the bridge response is quite slow, the controller's sample time is long enough to do those calculations.

3.5.3 Heat Pump On/Off Logic

As described in the previous section the heat pump's on command is determined by the meteorological controller. When there is a potential ice threat, the heat pump will be operated.

The heat pump is set "On" when there is a potential ice threat. The heat pump is set "Off" based on two conditions. First, if there is no future potential ice threat. Second, the Bridge Loop supply temperature after optimization calculation reaches its lower limit.

3.5.4 MPC Summary

Figure 3-21 summarizes the steps that make up the Smart Bridge MPC algorithm.

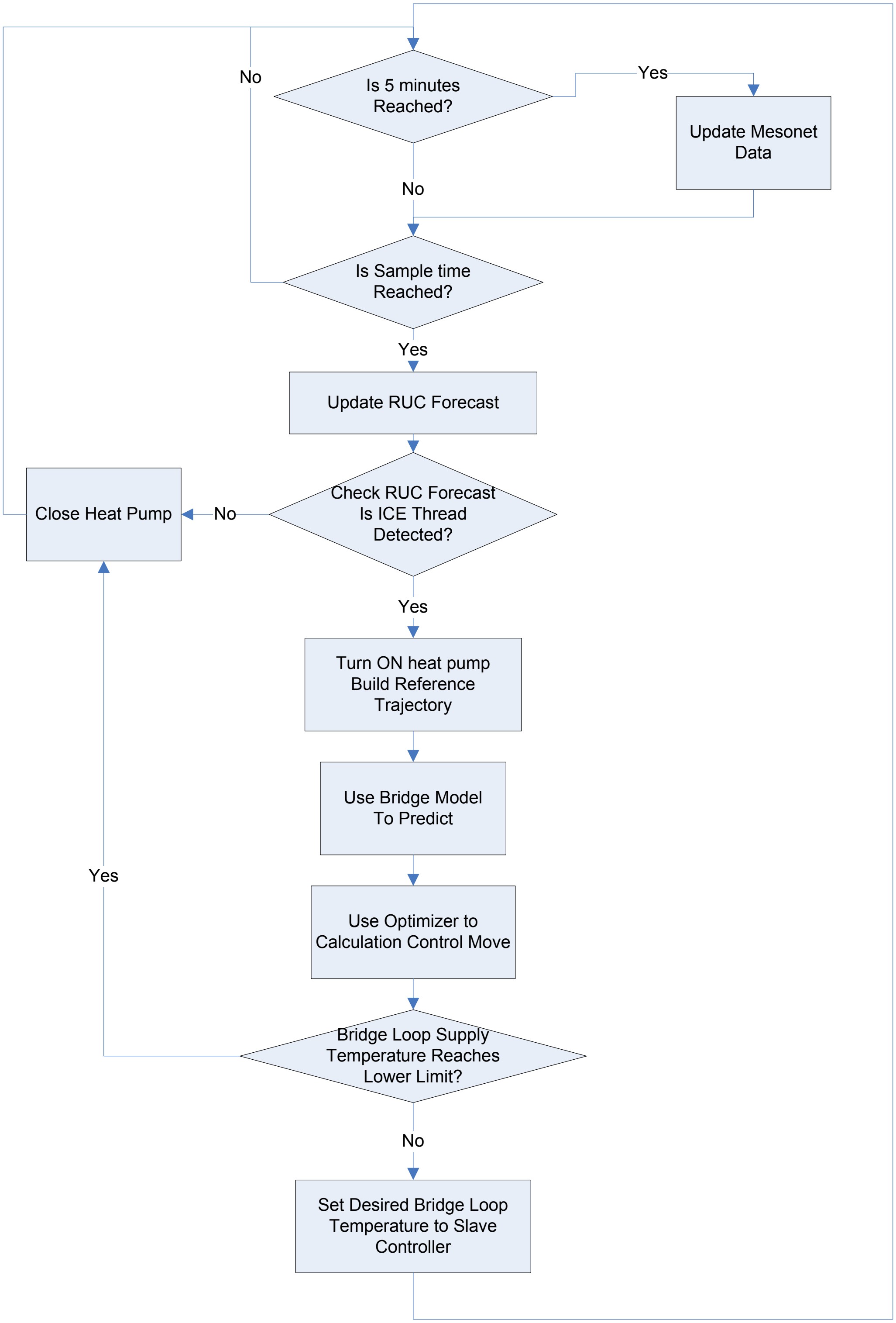


Figure 3-21: MPC Algorithm Summary

3.6 PID Controller (Slave)

The PID Controller is run on the Bridge PC. The heat pump is set “On” or “Off” by this PID controller based on the command from the MPC controller. The three way valve is manipulated by a conventional PID algorithm to control the Bridge Loop supply temperature. The MPC controller’s calculation result is served as the set point of the PID controller. Therefore, the MPC controller and the PID controller form a Master – Slave relationship.

The heat pump used on the bridge site is a Florida Heat Pump; model WP120, with a nominal capacity of 10 tons. Details can be found at: http://www.fhp-mfg.com/product_lines/wp.htm.

The three way valve is a HAYWARD EVS2 electric three way valve. Valve size is ¾”. Details can be found at www.haywardindustrial.com.

The Bridge Loop pipe’s diameter is 0.01905 meter (¾”). The total pipe length is 198.11 meters.

The digital PID controller is represented in velocity form.

$$\Delta u(k) = \kappa_c \{[\varepsilon(k) - \varepsilon(k-1)] + \frac{T}{\tau_i}\varepsilon(k) + \frac{\tau_d}{T}[\varepsilon(k) - 2\varepsilon(k-1) + \varepsilon(k-2)]\} \qquad \text{(Eq. 3-7)}$$

Where $\varepsilon = Setpoint - CV$ and $\Delta u(k) = u(k) - u(k-1)$

The PID Controller’s parameters are selected as follow:

$$\kappa_c = 0.006,\ \tau_i = 27\,\text{sec and } \tau_d = 6.4\,\text{sec}$$

The sample time (T) of the PID Controller is 10 seconds.

An output clamp is integrated in the PID algorithm. The maximum output change is 3% of the full range. This is based on the step test of the PID controller. Figure 3-22 and Figure 3-23 show the step test of the PID Controller on 11/20/2003 and 11/21/2003, with and without output clamp.

The PID Controller's set point was sequentially set to 20ºC, 30ºC, 40ºC and 50ºC and then set back in these two tests.

From Figure 3-22, without the output clamp, the controller is unstable. This is because the relation between the valve position and the Bridge Loop supply temperature is highly nonlinear.

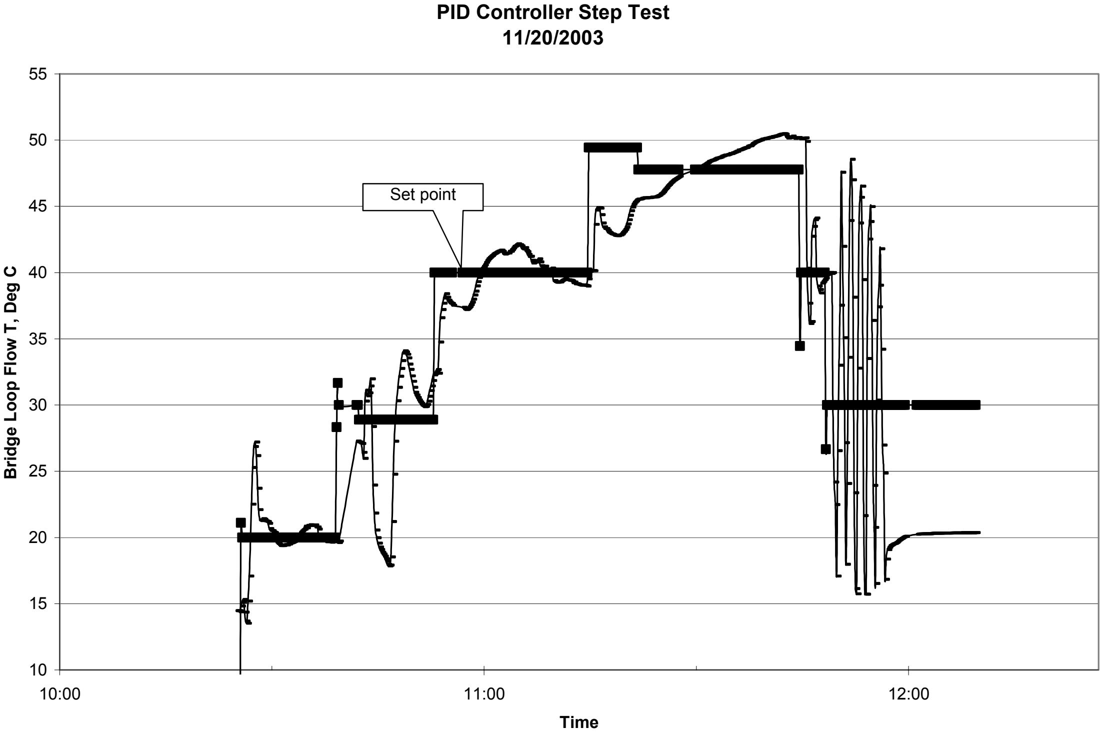


Figure 3-22: PID Controller Step Test – without Output Clamp

The Figure 3-23 shows by adding the output clamp logic, the PID Controller is stable. The output clamp eliminates the output oscillation.

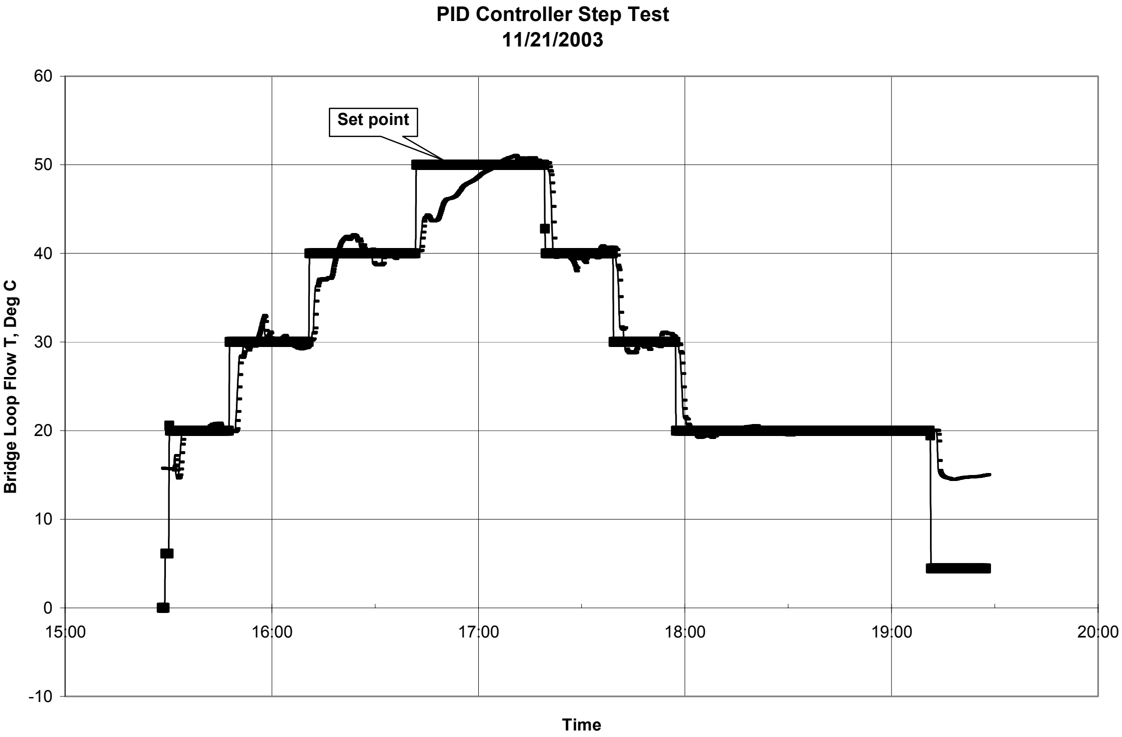


Figure 3-23: PID Controller Step Test – with Output Clamp

3.6.1 Bridge Site Measurements

There are several instruments installed on the bridge, such as flow meters, thermocouples, etc. Bridge site information, such as Bridge Surface temperature, Bridge Loop temperature, Bridge Loop flow rate, etc., is measured by these instruments.

There is a total of 112 points of bridge information that is gathered and recorded by a data acquisition program, which was written by other investigators working on the OSU Smart Bridge project, running on the Bridge PC. The need for such a large package of bridge information is because of other investigators' research on the Smart Bridge Project.

These 112 points of information is sent to the PID Controller software by the Windows DDE mechanism as a data array. The PID controller only uses parts of this information. Table 3-6 lists the data points used by the PID controller. In detail, the data array has 112 elements; offset xx means the number xx element in the array.

Offset in the Array	Information Description
1 ~ 20	Point Bridge Surface Temperature
64	Air Temperature near Bridge
69	Bridge Loop Flow Outlet Temperature
70	Bridge Loop Flow Inlet Temperature
82	Bridge Loop Flow Rate

Table 3-6: Bridge Information Used By the PID Controller

The point 1~20 Bridge Surface Temperatures are measured by 20 thermocouples embedded 0.375 inches below the bridge pavement surface. As described in Chapter II, only half of the OSU test bridge is equipped with the heating system. These thermocouples are embedded in the heated part of the bridge.

Figure 3-24 shows the distribution of these thermocouples on the OSU Test Bridge. By using the south west corner of the bridge as the origin, each thermocouple's coordinator is listed in Table 3-7.

Currently, the average bridge surface temperature is the average of the measurements of interior thermocouples (T1, T2, T5, T6, T9, T10, T13, T14, T17, and T18).

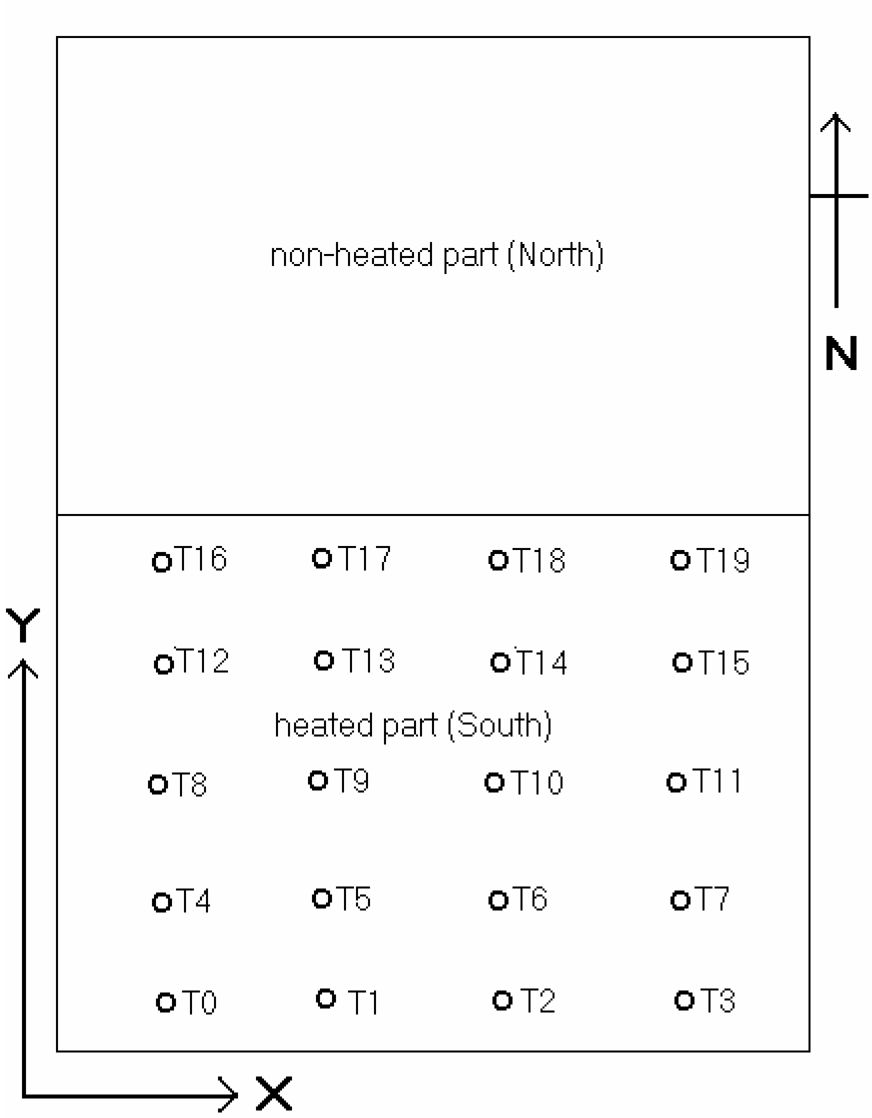


Figure 3-24: OSU Test Bridge Thermocouple Grid

Sensor	X-Coordinate (inches)	Y-Coordinate (inches)
T0	35.15671	38.53925
T1	77.17028	32.3474
T2	155.36444	29.92704
T3	200.26479	36.37484
T4	35.75007	120.2096
T5	78.72011	115.04794
T6	154.69591	114.26261
T7	200.7227	120.74948
T8	37.11788	181.1627
T9	80.37086	173.48996
T10	154.53102	172.73781
T11	199.26384	179.37157
T12	36.90292	241.89908
T13	78.91352	235.54297
T14	154.48934	234.28106
T15	200.94865	239.3743
T16	36.8344	321.99261
T17	80.8579	316.23037
T18	154.91064	315.86026
T19	201.33461	319.74854

Table 3-7: Thermocouple Coordinates

3.7 Programming Notes

Several programming languages were used in writing the Smart Bridge control system's software, including Visual Basic, C++, Delphi and Assembly Language. No matter what kind of language is used, the object oriented and modular principles are always integrated into the software development.

The MPC Controller was programmed in Visual Basic using an Object Oriented Programming approach. The core controller logic was implemented in an object oriented format. Each object is a combination of methods and members. The objects have been carefully designed with only the necessary methods exposed to its peers. The encapsulation nature of the objects has dramatically improved the modularity of the software and reduced the models' interdependence. This makes the software more robust, easy to debug and flexible to future changes and enhancements. Currently, the objects in the software are: Controller, StepModel, BridgeModel, Mesonet, RUC, Communication, Display, Log, and File (Figure 3-25, ellipse). Each object provides specific functionality that can be utilized by any other object in the composite system. This modular structure is essential to support the varying sources and types of information that will be provided at different bridge sites.

See Appendix A for detailed information of the objects used in the software.

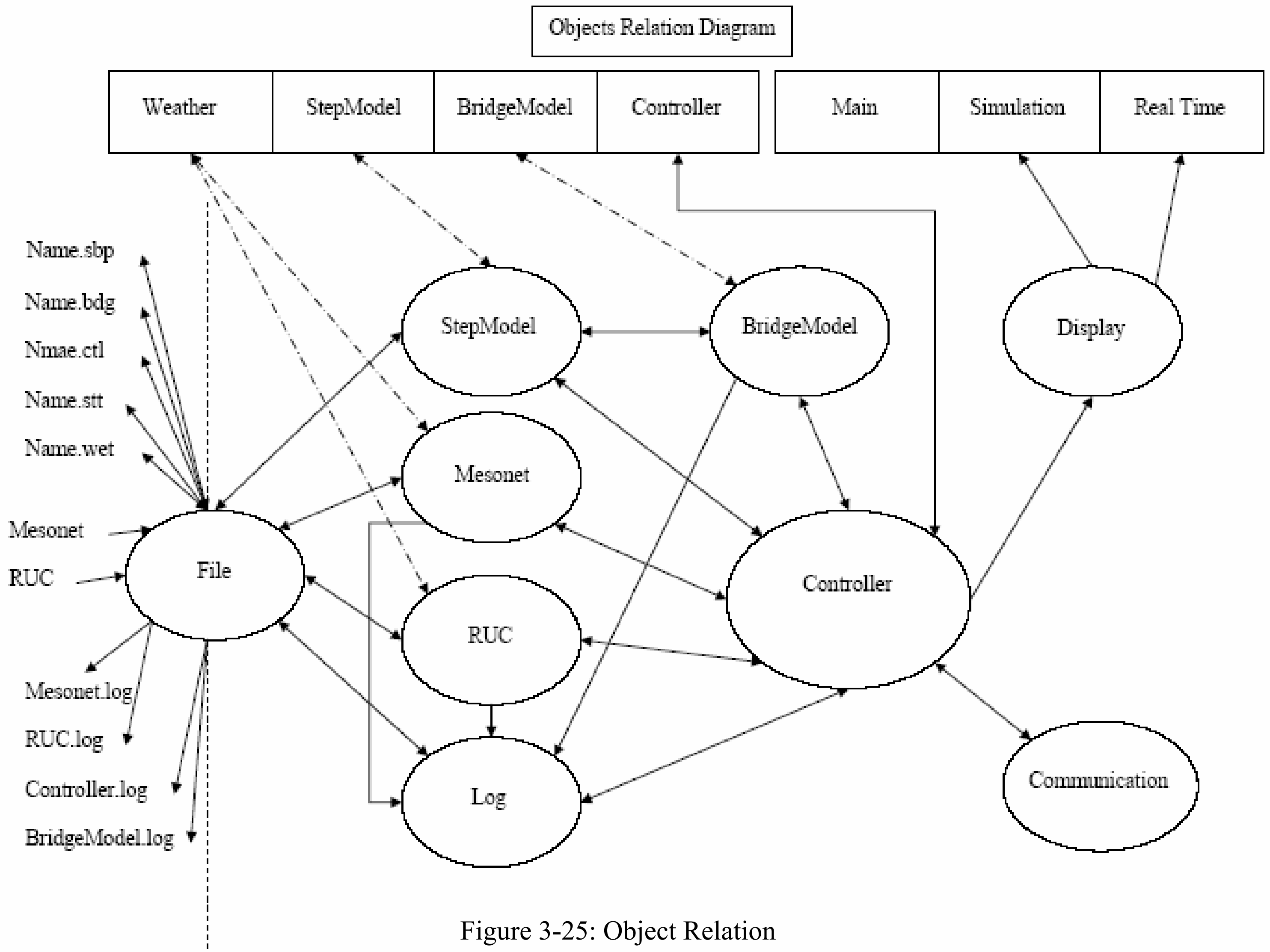
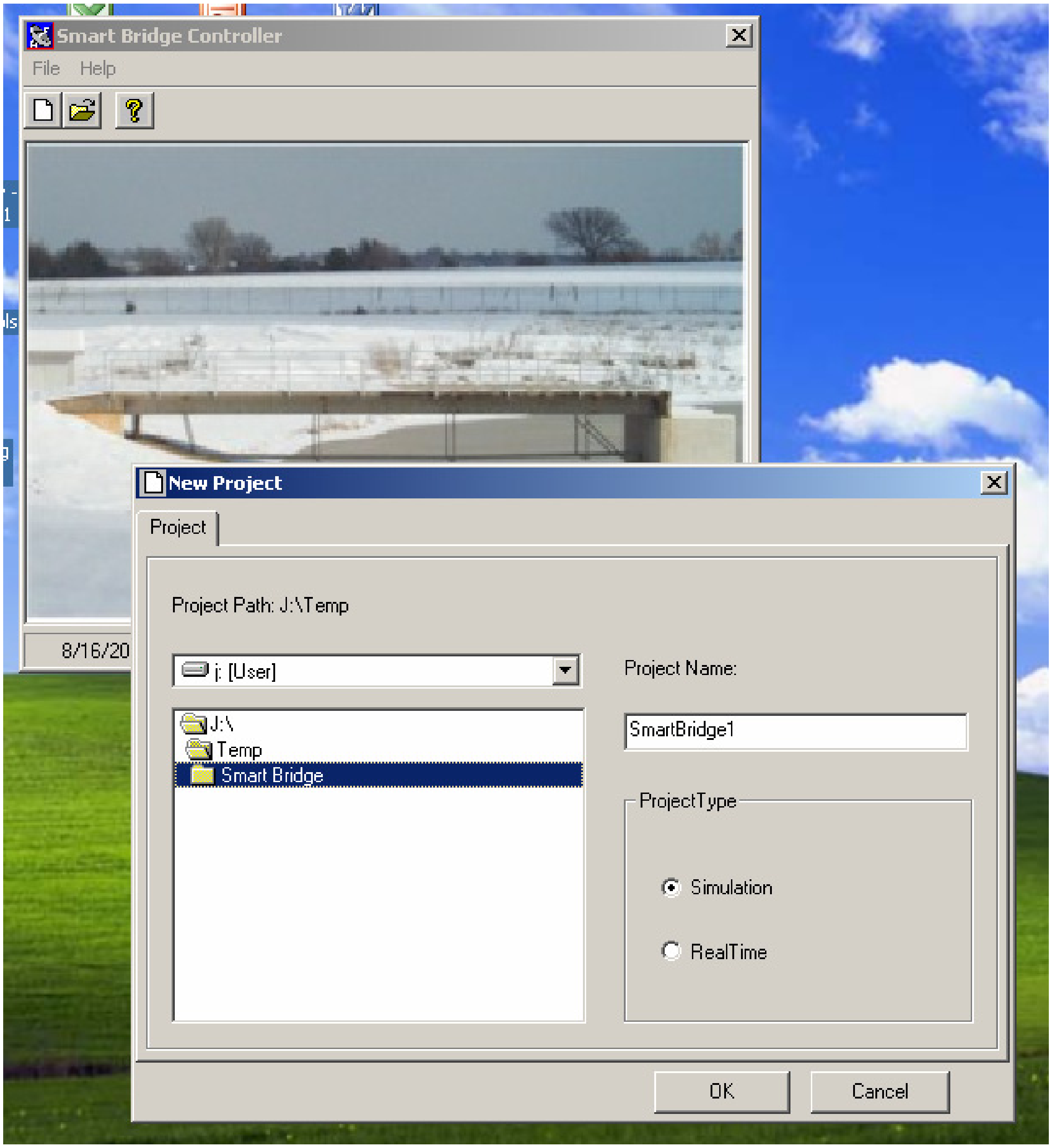


Figure 3-25: Object Relation

A project concept has been introduced to software. A project is a structured collection of files containing bridge parameters, controller configuration information, weather measurements, weather forecasts, and historian files. A project provides all of the documentation necessary to assess or recreate controller performance. In Figure 3-25, all the files on the left part, interacting with the File object are part of the project. See Appendix B for detailed information of the project concept.

All the rectangles in Figure 3-25 are user interface. Figure 3-26 to 3-28 shows some selected screen captures of the software Graphical User Interface (GUI). See Appendix C for all the user interfaces.



Project Path: J:\Temp

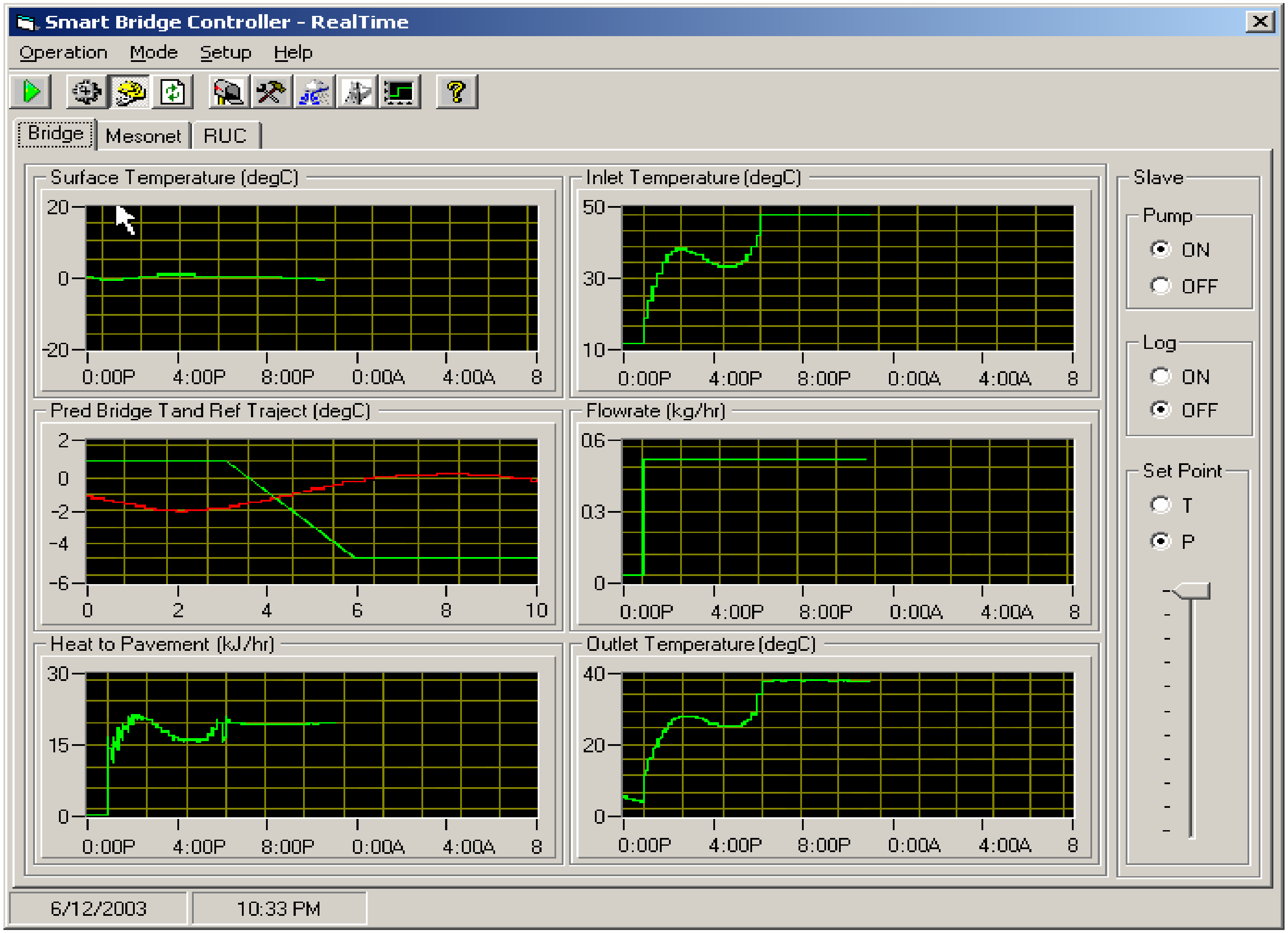
Project Name: SmartBridge1

ProjectType

● Simulation

○ RealTime

Figure 3-26: Project Selection



Pump: ● ON ○ OFF

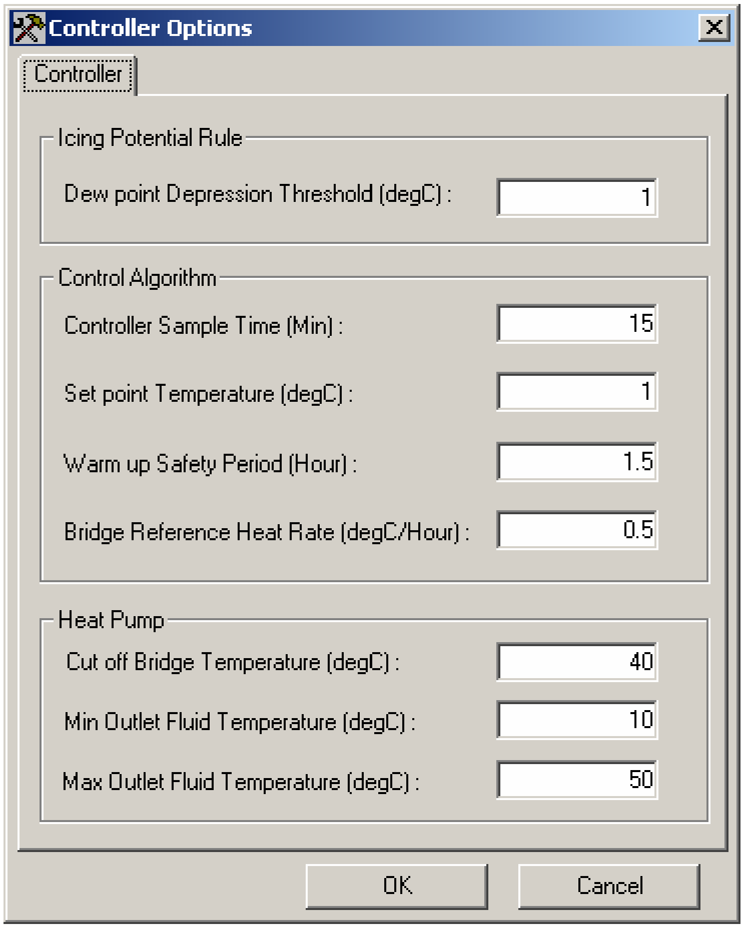
Log: ○ ON ● OFF

Set Point: ○ T ● P

6/12/2003

10:33 PM

Figure 3-27: Real Time Interface



Icing Potential Rule

Dew point Depression Threshold (degC) : 1

Control Algorithm

Controller Sample Time (Min) : 15

Set point Temperature (degC) : 1

Warm up Safety Period (Hour) : 1.5

Bridge Reference Heat Rate (degC/Hour) : 0.5

Heat Pump

Cut off Bridge Temperature (degC) : 40

Min Outlet Fluid Temperature (degC) : 10

Max Outlet Fluid Temperature (degC) : 50

Figure 3-28: Controller Parameter Input Dialog

CHAPTER IV

CASE STUDY PART I – REAL TIME RESULTS

4.1 Introduction

The objective of the Smart Bridge control system is to develop an automatic control system that maintains ice-free bridge conditions while minimizing heated bridge life-cycle cost. Previous chapters have described the advanced weather forecast and the MPC technique utilized by the Smart Bridge control system. These techniques represent the basis of the system.

During winter 2003-2004, the control system was tested on the OSU medium-scale test bridge, located on the west campus of Oklahoma State University. This chapter presents the system performance results.

Four snow events were recorded: (1) December 9, 2003, (2) December 12, 2003, (3) January 26, 2004, and (4) February 4, 2004. Sections 4.2 to 4.4 present the results and analyses for each of these events. Each case study begins with a description of the weather condition associated with the event. The controller's settings are then listed. The control performance results are then presented and analyzed.

4.2 Snow Event 1, December 9, 2003

The ambient air temperature for snow event 1, December 9, 2003, is shown in Figure 4-1. The air temperature was below 0°C during the entire period. According to the observation record, rain started at 12:00, December 9, developed into snow at about 16:30, December 9, and ended at about 22:00, December 9.

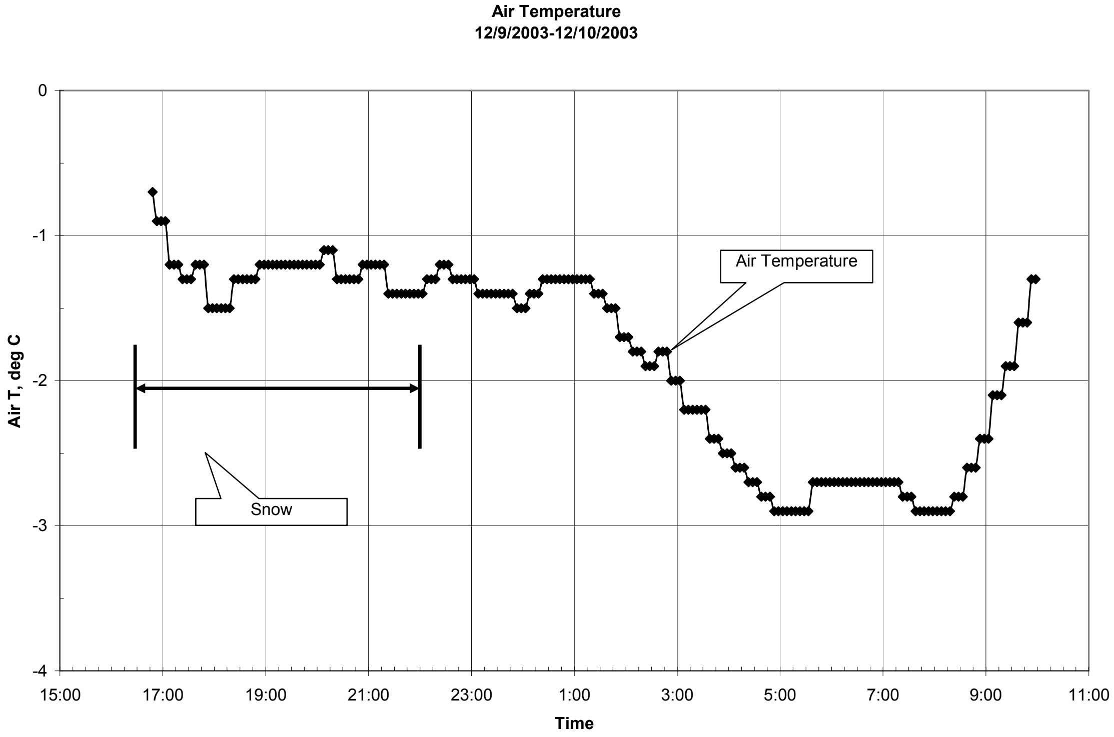


Figure 4-1: Event 1 Air Temperature

The controller parameters were set as follows: Dew Point Depression Threshold = 3°C; Controller Execution Period = 15 minutes; Bridge Deck Set Point Temperature = 3°C; Warm up Safety Period = 1 hour; Bridge Reference Heat Rate = 2°C/hr. The settings mean the bridge surface temperature should be heated to 3°C at least one hour before the predicted ice threat.

However, the RUC forecast feed was lost at 05:00 in the morning and not restored until 16:50. The loss of forecast information prevented the controller from preheating the bridge prior to the start of the snow event.

The controller turned on the heating system immediately after the RUC data stream was restored at 16:50 (Figure 4-2). The controller kept the heating system on during the rest of the snow event and turned off the heating system at 02:28 on 12/10/03, approximately 4.5 hours after the snow storm ended.

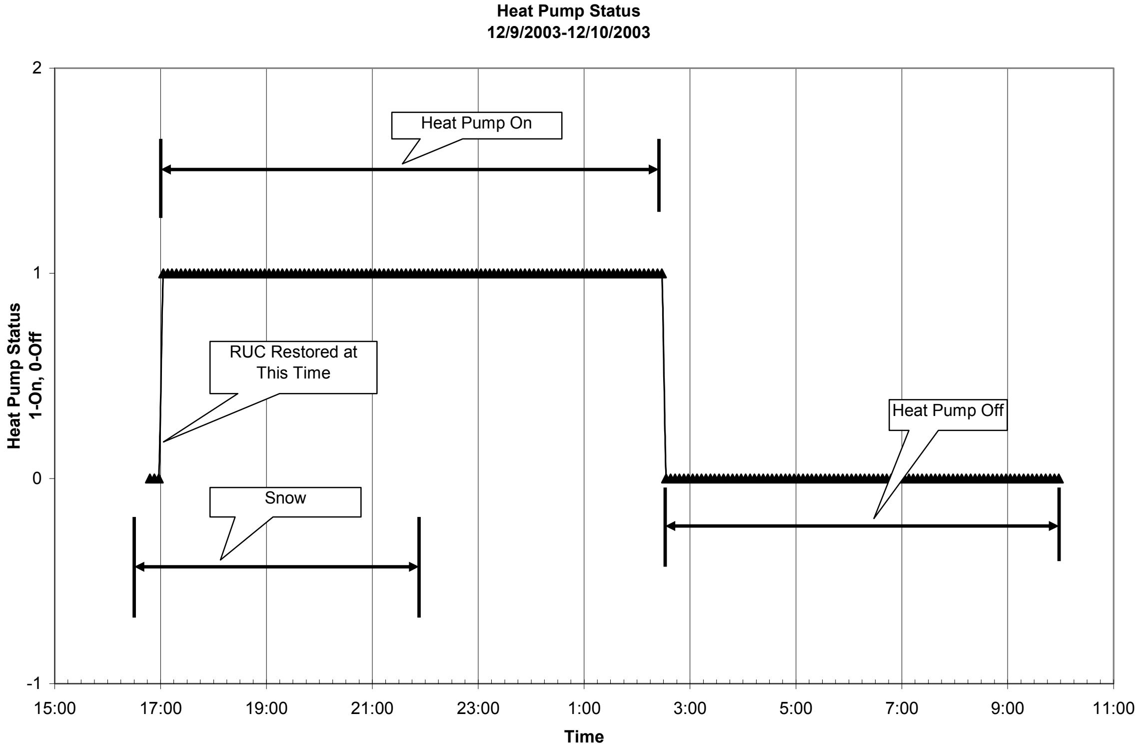


Figure 4-2: Event 1 Heat Pump Status

Figure 4-3 shows the bridge surface temperature trend during this snow event. The 3°C line in Figure 4-3 is the desired bridge surface temperature.

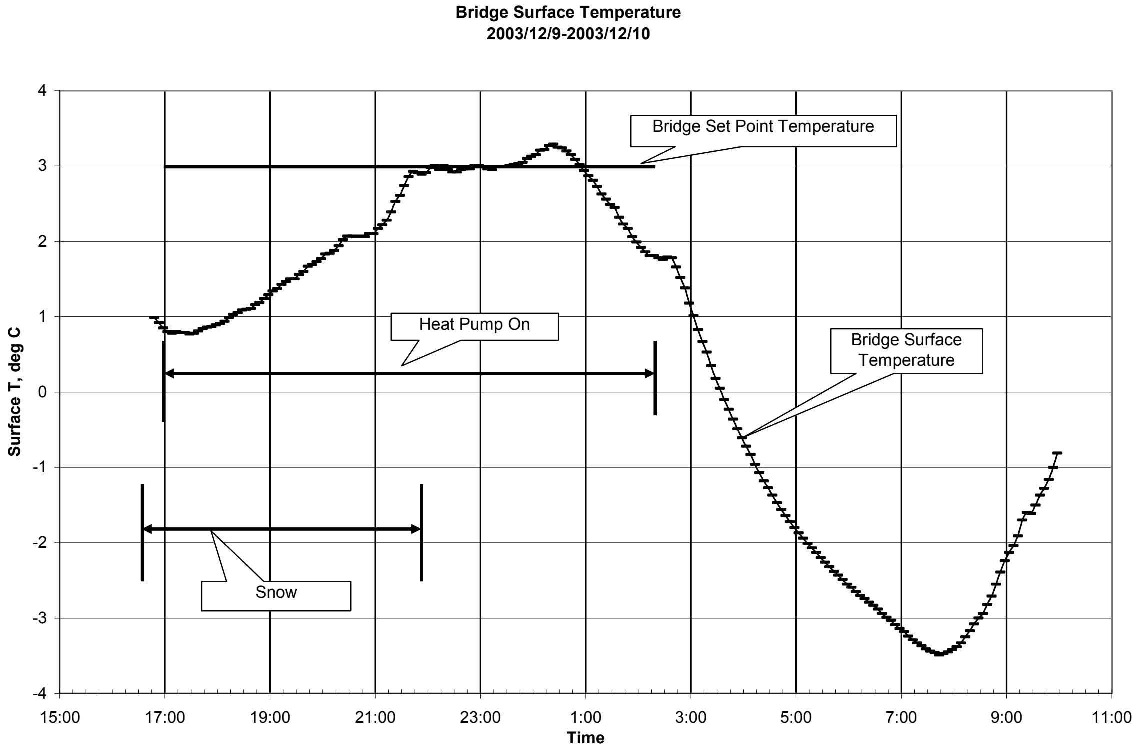


Figure 4-3: Event 1 Bridge Surface Temperature

The bridge surface temperature started increasing after the heating system was engaged and reached the desired bridge temperature at about 22:00. Approximately five hours were required after the heating system was first engaged to raise the bridge surface temperature to the desired value.

Figure 4-4 shows the Bridge Loop temperature set point record. These were the control actions calculated by the MPC controller. They were sent to the PID controller, serving as the set point for the Bridge Loop supply temperature.

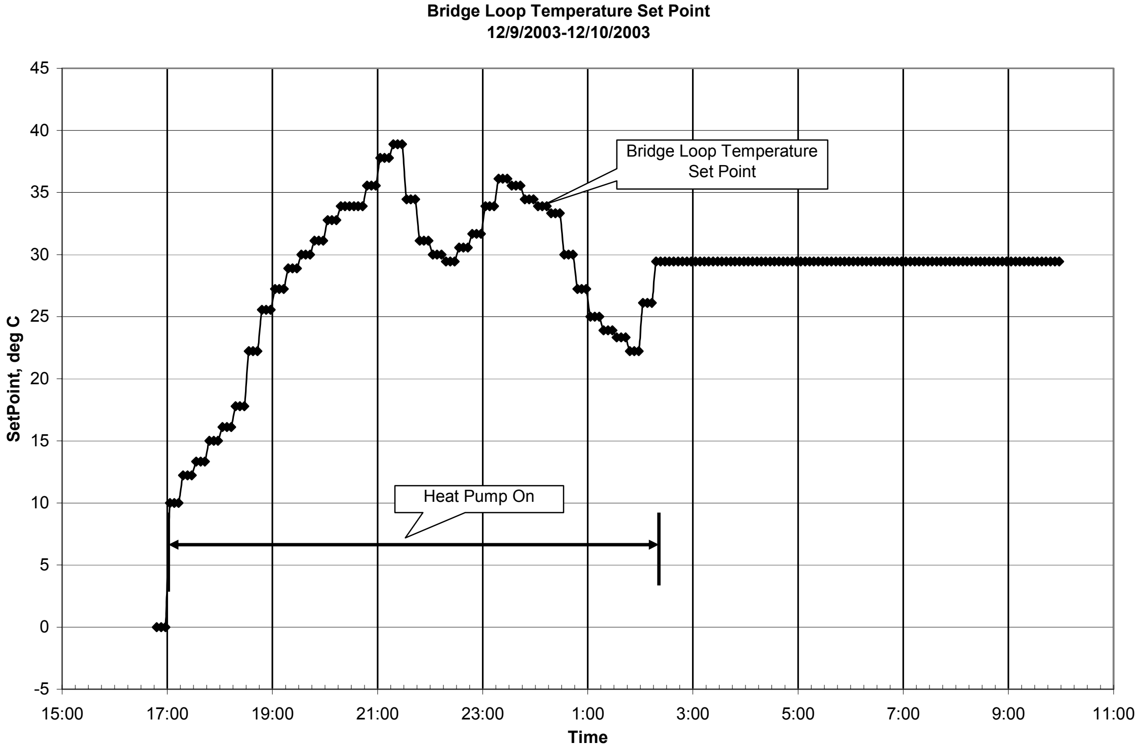


Figure 4-4: Event 1 Bridge Loop Temperature Set Point

Figure 4-5 shows the actual Bridge Loop supply temperature trend during the snow event. When the heating system was engaged, the inlet supply temperature followed the set point quite well. This confirms the PID controller worked well.

During the snow event, the bridge surface temperature was maintained above 0°C. However, undesirable controller behavior was noted. From 17:00 to 21:00, though the bridge surface temperature was far below the desired value, instead of using the full capacity of the heating system, the MPC controller raised the Bridge Loop temperature set point steadily but slowly. Because of this, it took longer than necessary for the bridge surface temperature to reach the desired value. The MPC control action was sluggish.

After the bridge surface temperature reached the desired value, the MPC controller worked quite well and maintained the bridge surface temperature within 1°C of the set point.

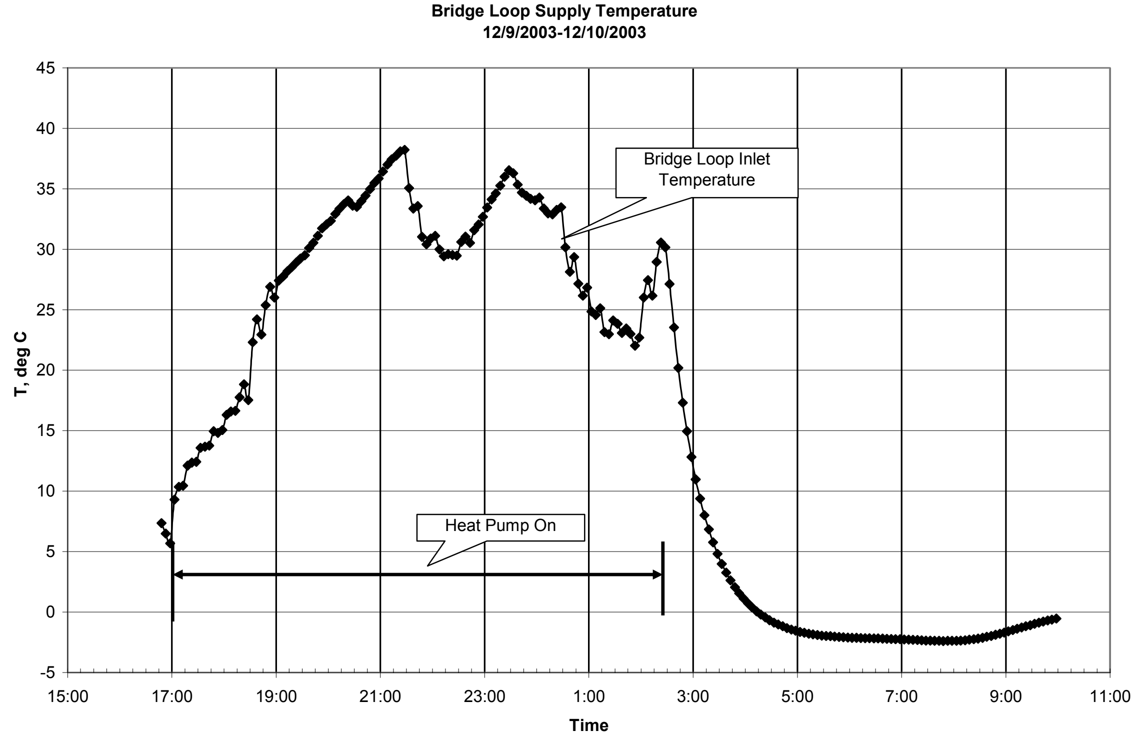


Figure 4-5: Event 1 Bridge Loop Supply Temperature

Another interesting result was also noticed in this snow event. The snapshot in Figure 4-6 was taken at 19:38. At that time, the average surface temperature for the heated portion of the bridge deck was reported to be 1.5°C. Though the temperature was above 0°C, snow accumulation occurred.

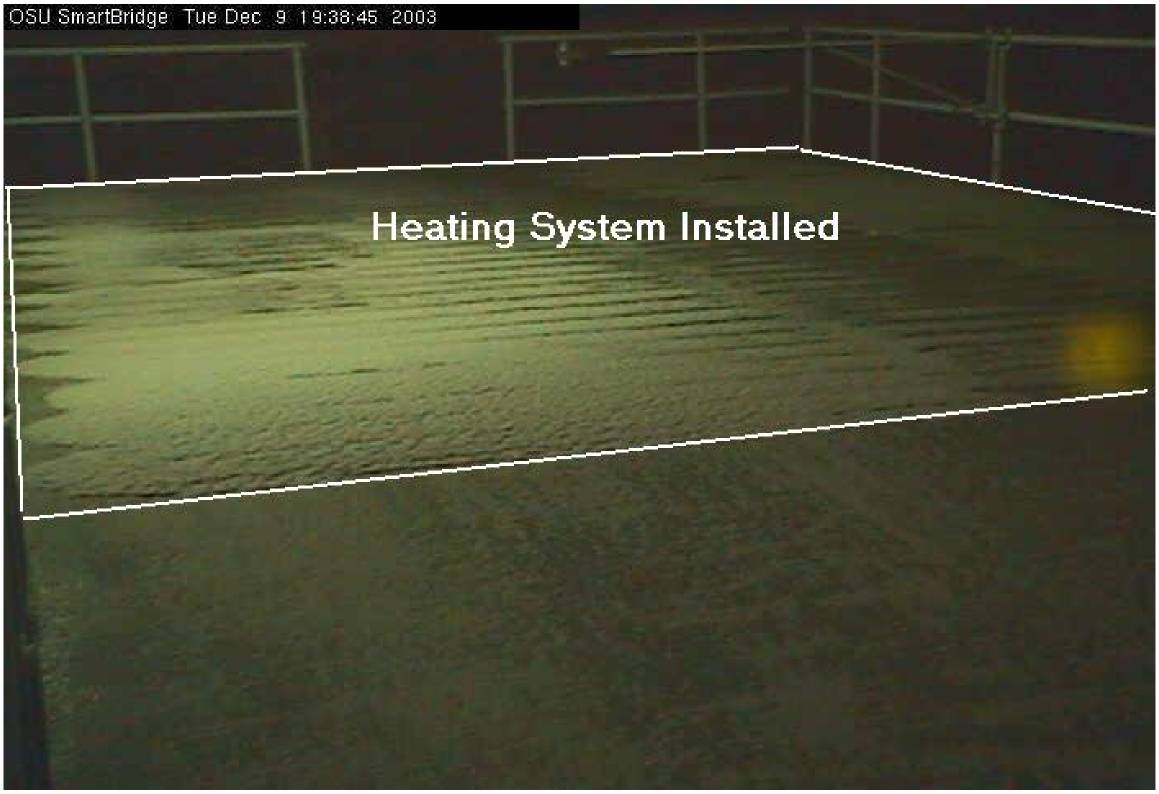


Figure 4-6: Event 1 Bridge Snapshot at 19:38, 12/9/2003
(Snow, +3 hours)

Suspected reasons for this behavior are as follows. First, the snow was preceded by rain. The bridge surface was wet before the snow. After the snow began, because of the operation of the heating system, the surface temperature of the heated area was kept above 0°C. Therefore, the heated bridge deck surface remained wet. On the other hand, the surface temperature of the non-heated area was below 0°C, and the rain had turned into ice. Snow adheres more easily to a wet surface than to ice.

However, snow still can provide more friction than ice to the vehicles. The area with heating system installed is still safer than the non-heating area.

Figure 4-7 shows the snapshot at 23:39, December 9, 2003. The heating system had been engaged for about 6.5 hours. Almost all the snow had been melted. The non-heated area remained covered by ice.

Figure 4-7: Event 1 Bridge Snapshot at 23:39, 12/9/2003
(Snow, +7 hours)

Conclusion**:** The controller worked correctly. It turned on the heating system right after the RUC was recovered and maintained the bridge surface temperature within 1°C of the set point. The PID Controller worked great. The Bridge Loop supply temperature tracked the set point perfectly. The fact that the preheating was not as quick as possible pointed out the controller needed some tuning.

4.3 Snow Event 2, December 12, 2003

The ambient air temperature for snow event 2, December 12, 2003, is shown in Figure 4-8. The air temperature was below 0°C 95% of the period. According to the observation record, rain started at 13:30, December 12, developed into snow at about 15:30 and ended at about 23:30, December 12, 2003.

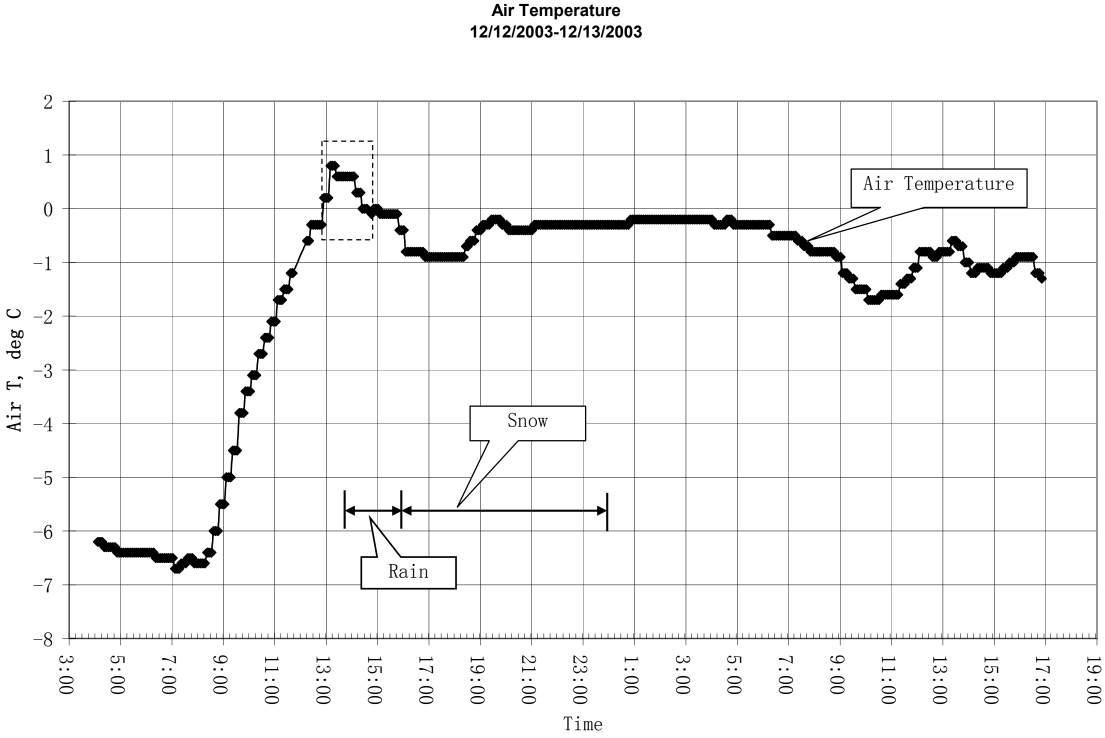


Figure 4-8: Event 2 Air Temperature

The controller parameters were set as follows: Dew Point Depression Threshold = 3°C; Controller Execution Period = 15 minutes; Bridge Deck Set Point Temperature = 4°C; Warm up Safety Period = 2 hour; Bridge Reference Heat Rate = 0.5°C/hr. The settings mean the bridge surface temperature should be heated to 4°C at least two hours before the predicted ice threat.

The controller turned on the heating system at 5:37, December 12, about 8 hours before the rain began (Figure 4-9). This control decision to turn on the heating system was based on the dew point depression rules as RUC forecast did not include rain/snow.

The bridge surface temperature increased quickly. At 9:00, December 12, the bridge surface temperature first reached 4°C (Figure 4-10), the desired value. At 9:37, the controller turned off the heating system as the RUC forecast indicated the bridge deck temperature would remain above set point due to ambient conditions.

From 9:37 to 12:30, December 12, the bridge was unheated. Bridge surface temperature was remained above 4°C (Figure 4-10). This confirmed the controller model prediction.

At 12:35, December 12, the control engaged the heating system again. This time the decision was based on both the dew point depression and RUC forecast precipitation. However, because the bridge surface temperature was already above the desired value, the MPC controller generated a Bridge Loop temperature set point near the lower limit of 10°C. Eventually the controller shut down the heating system when the set point fell below the lower limit at 13:50, December 12.

From 14:00 to 15:00, December 12, the air temperature remained above 0°C. The RUC update predicted the air temperature would go higher (Figure 4-12). Based on this forecast, the MPC controller predicted a further increase in the bridge surface temperature. Therefore, the heat pump was kept off during this time period. However, the weather forecast was incorrect, the air temperature decrease. Rain began at 13:30, December 12. Without heating, the bridge surface dropped.

At 15:01, December 12, the heating system was re-engaged. As described for the previous event, the controller increased the Bridge Loop temperature set point slowly (Figure 4-11). It took approximately five hours for the bridge surface temperature to return to the desired value.

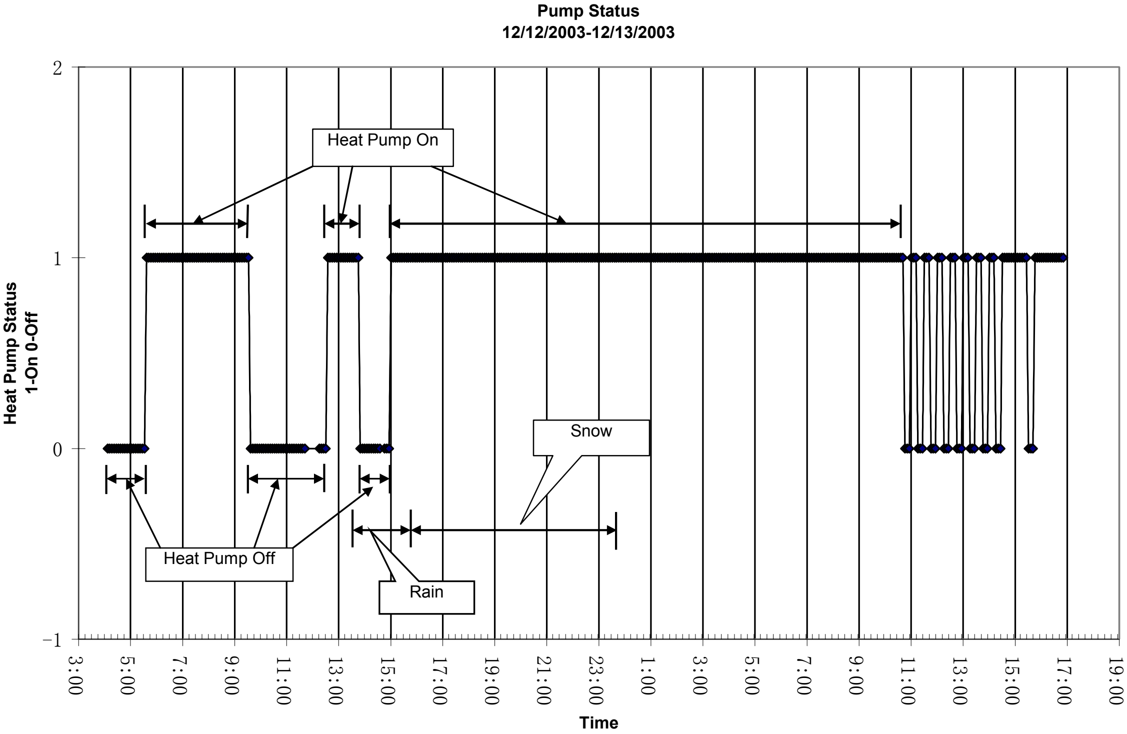


Figure 4-9: Event 2 Heat Pump Status

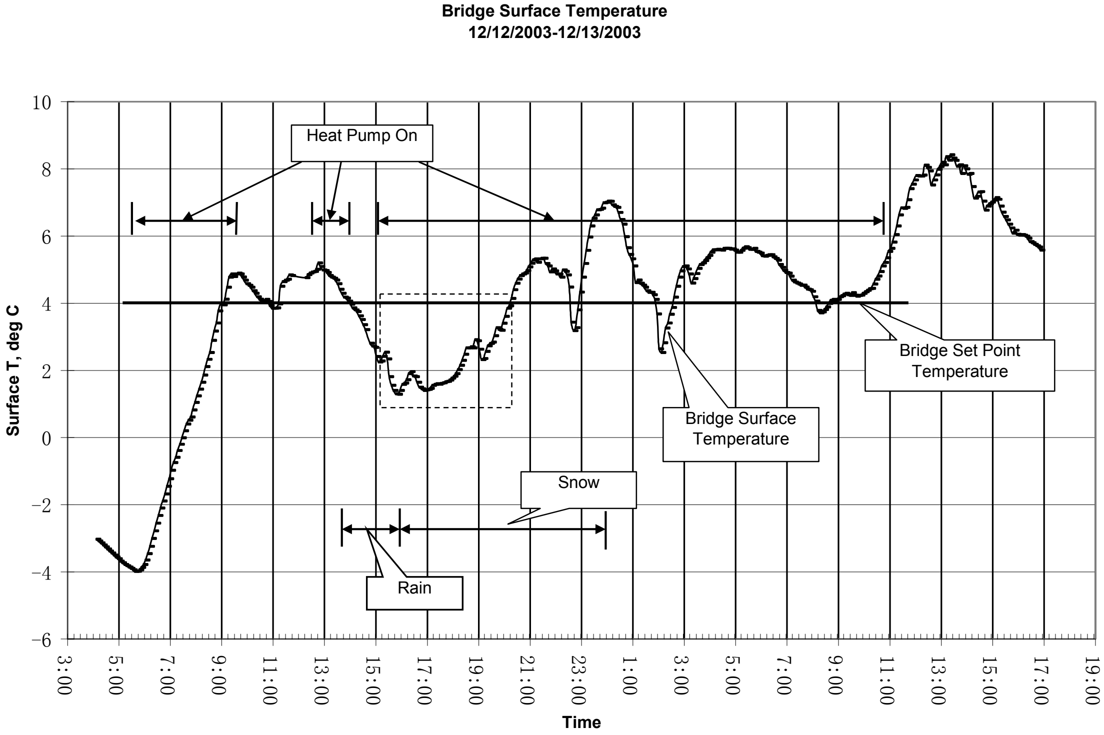


Figure 4-10: Event 2 Bridge Surface Temperature

The heating system remained on until 11:00, December 13. The heating system then cycled on and off several times. This occurred in response to RUC forecast of approaching ice events. The heat pump was turned on based on this condition. However, the bridge surface temperature was well above the desired set point. In response the controller generated a set point value below the lower cutoff limit. Therefore, the heat pump was turned off (refer to the MPC algorithm described in Chapter III). This sequence repeated several times until the bridge surface temperature decreased closer to the desired set point.

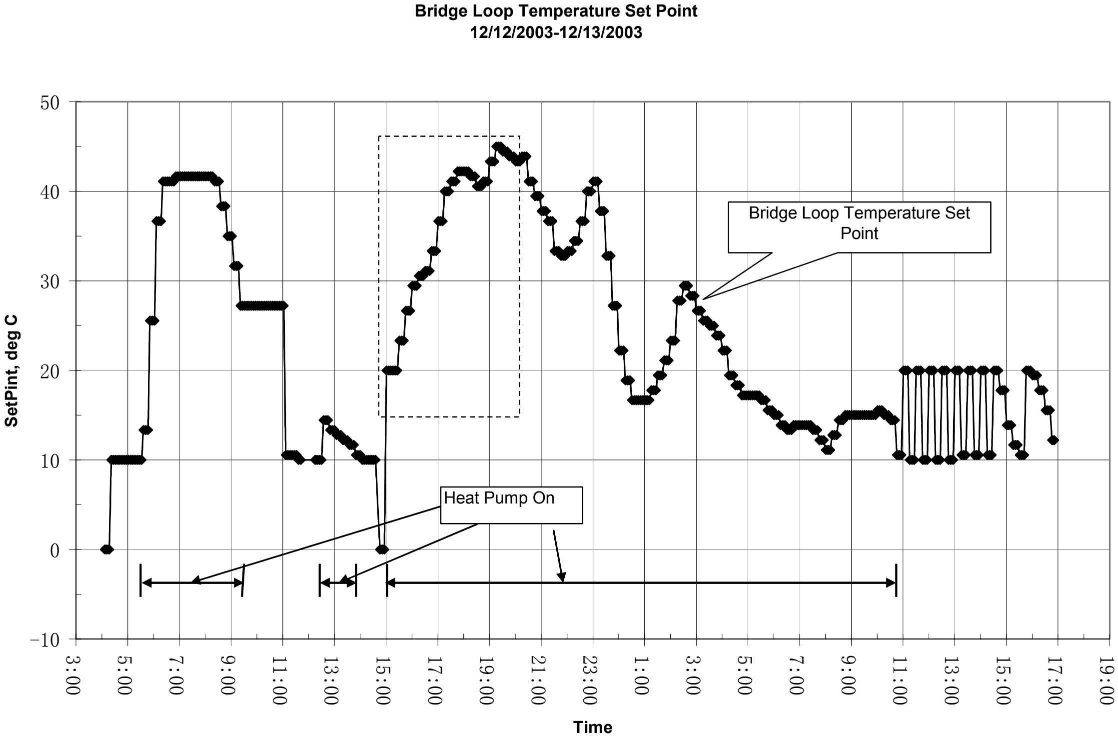


Figure 4-11: Event 2 Bridge Loop Temperature Set Point

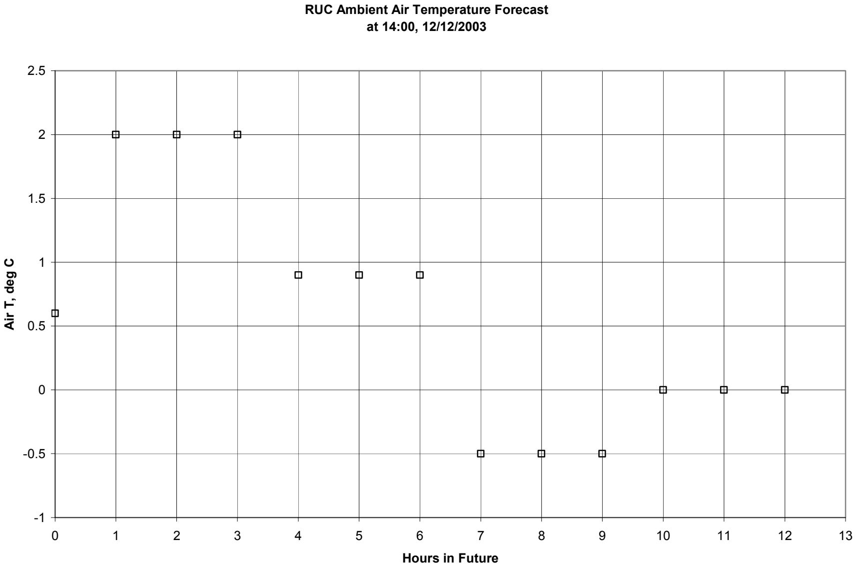


Figure 4-12: Event 2 RUC Air Temperature Forecast at 14:00, 12/12/2003

With regard to the actual precipitation, the control system started heating the bridge eight hours before the potential ice threat. The desired temperature was achieved three hours before the rain. This is quite good. Though the bridge temperature was not under control for about one hour during the rain, no ice was formed during the rain and no snow accumulation during the first hour of the snow.

Figure 4-13 shows the bridge surface condition at 17:17. Snowfall began about an hour earlier. There was almost no snow accumulation on the heated area, while the unheated part of the bridge was already covered by snow.

Between 15:00 and 20:00, December 12, though the heating system was engaged, the controller raised the Bridge Loop temperature set point quite slow (Figure 4-11), consequently, the bridge surface temperature remained below set point for five hours. The bridge got some snow accumulation after 17:30. However, with the bridge surface temperature restored, the snow was melted very quickly. At 20:17, the heated area of the bridge was almost clear (Figure 4-14).

Conclusion: The feasibility of the advance weather forecast and MPC algorithm was demonstrated by the performance of the control system. The control system detected the on-coming ice event and preheated the bridge in the proper manner. Results confirm that the need to tune the controller for more aggressive behavior during the bridge preheat period.

Figure 4-13: Event 2 Bridge Snapshot at 17:17, 12/12/2003
(Snow, +2 hours)

Figure 4-14: Event 2 Bridge Snapshot at 20:17, 12/12/2003
(Snow, +5 hours)

4.4 Snow Event 3, January 26, 2004

Prior to this event, the controller was adjusted to increase the preheat rate. All the diagonal elements of the Λ matrix were changed from 1.0 to 0.5.

The ambient air temperature for snow event 3, January 26, 2004, is shown in Figure 4-15. It was extremely cold that day. The air temperature kept dropping during the entire period.

According to the observation record, sleet started about 13:30, January 26 and ended at about 16:30, January 26. Wind was hard during the sleet.

The controller parameters were set as follows: Dew Point Depression Threshold = 3°C; Controller Execution Period = 15 minutes; Bridge Deck Set Point Temperature =

4°C; Warm up Safety Period = 2 hour; Bridge Reference Heat Rate = 0.5°C/hr. The settings mean the bridge surface temperature should be heated to 4°C at least two hours before the predicted ice threat.

Unfortunately, the RUC forecast was lost again until 13:00. This prevented the controller from preheating the bridge.

The controller turned on the heating system immediately after the RUC forecast was restored and kept the heating system on until midnight.

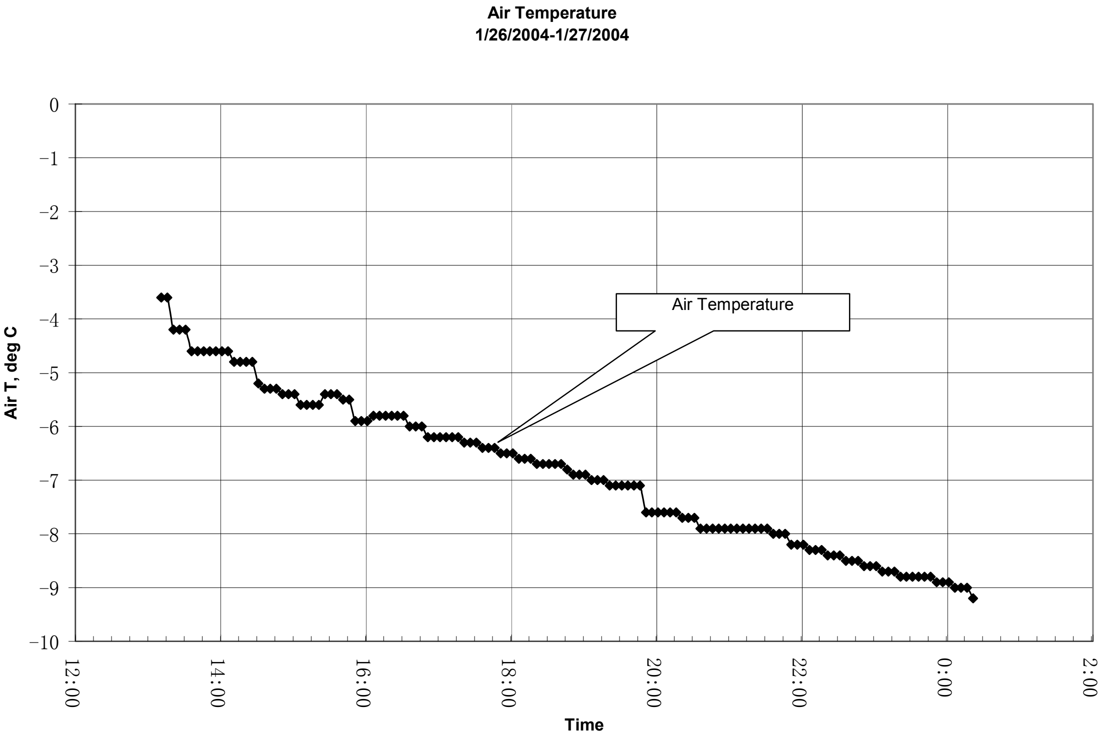


Figure 4-15: Event 3 Air Temperature

Although the controller called for maximum heating (Bridge Loop supply temperature set point = 50°C), the weather condition were so severe that the actual Bridge Loop supply temperature could only reach 37°C (Figure 4-17). Because the heat losses exceeded the heating capacity of the system, the bridge surface temperature could not be maintained at set point during this snow event. The bridge deck increased to 1°C at about 15:00, and then dropped continuously after 16:00 (Figure 4-18).

Despite the controller tuning adjustment made prior to this event, the controller still preheated the bridge at less than the maximum rate possible. This indicates the controller adjustment was not effective.

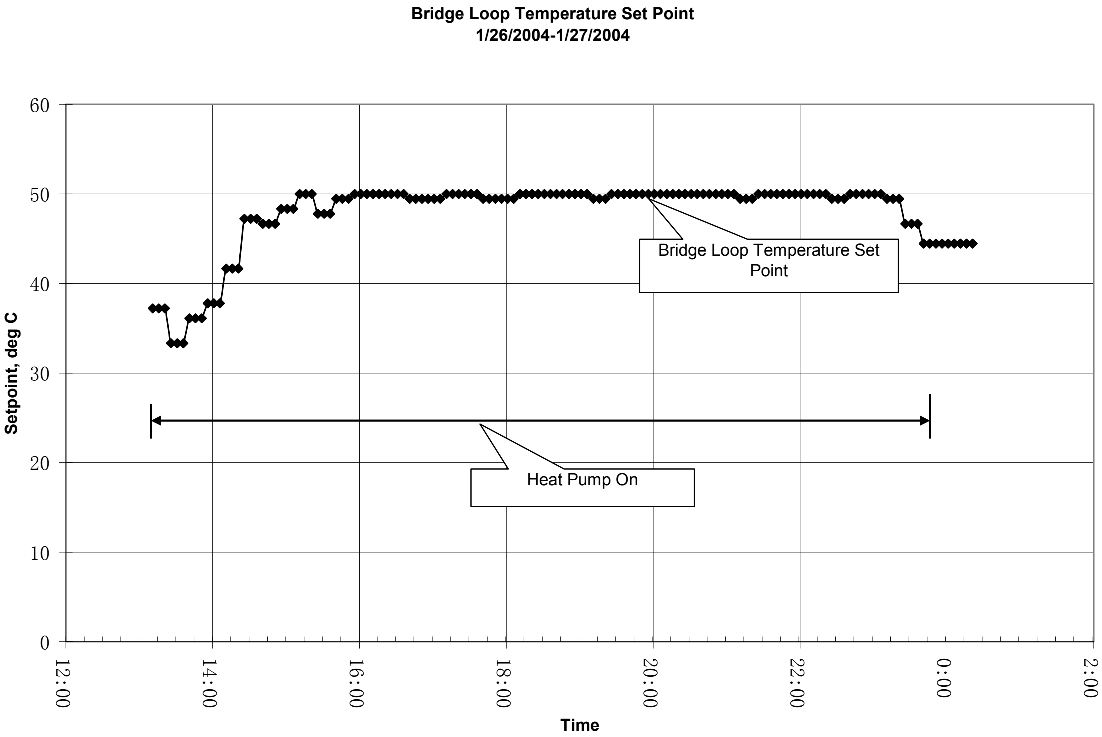


Figure 4-16: Event 3 Bridge Loop Temperature Setpoint

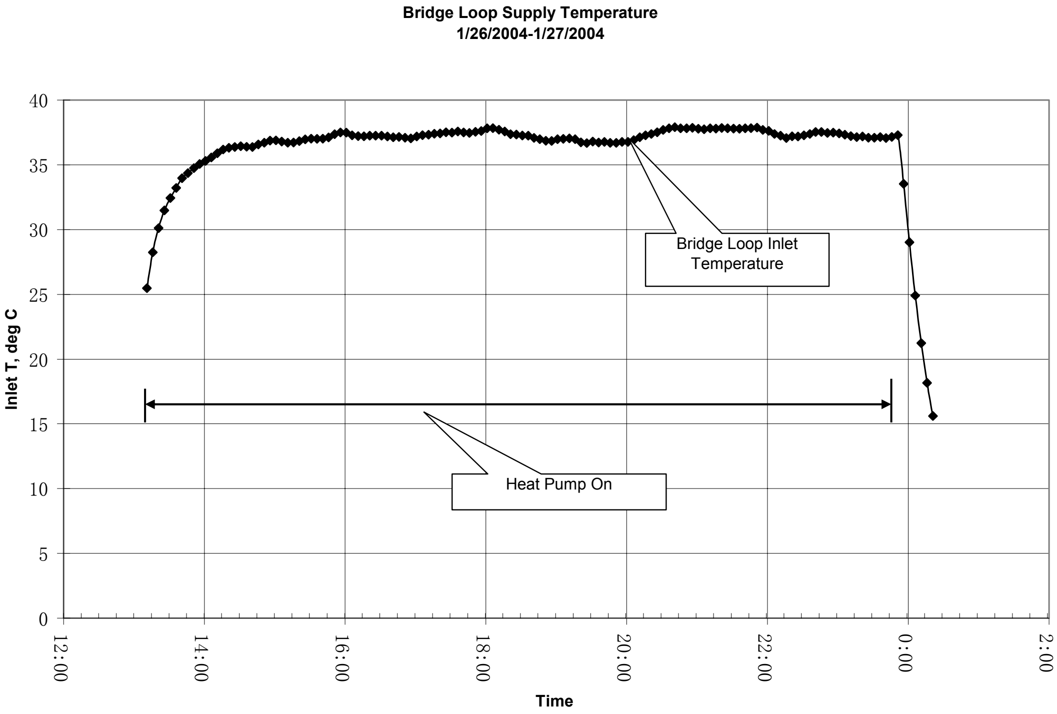


Figure 4-17: Event 3 Bridge Loop Supply Temperature

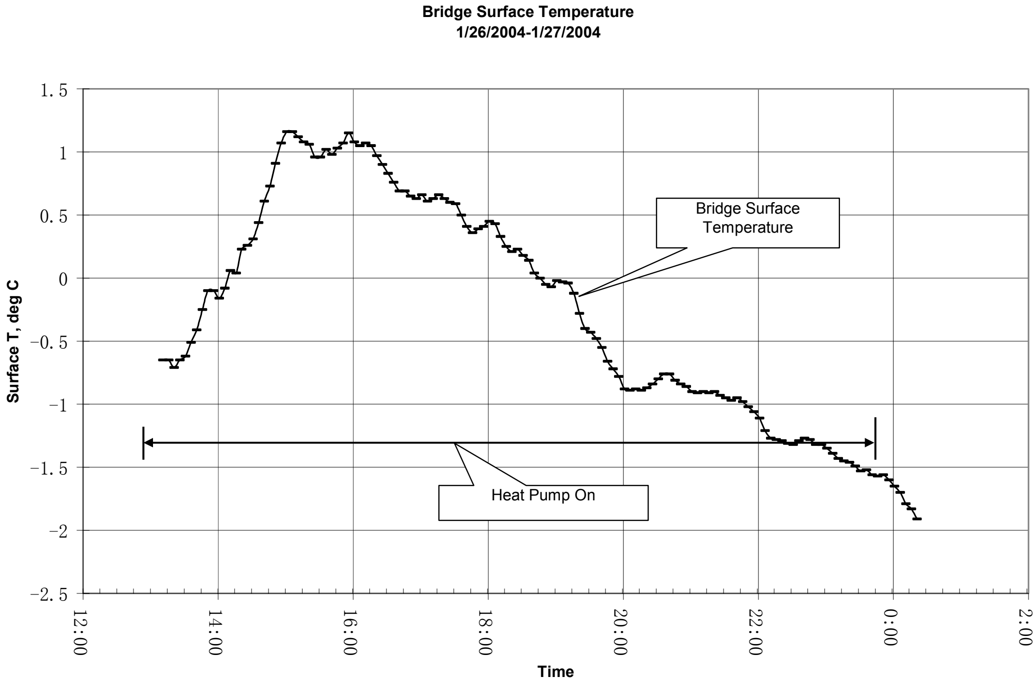


Figure 4-18: Event 3 Bridge Surface Temperature

Conclusion: From a control engineering viewpoint, operating the heating system at maximum capacity means that the system is not under control. This snow event indicated that the heating system for the OSU test bridge is undersized. It may be possible to partially compensate by preheating the bridge in advance.

The RUC forecast is essential to the controller. Without the forecast, the preheat procedure can not be performed correctly.

The modified tuning of the MPC matrix weight did not have much effect.

4.5 Snow Event 4, February 4, 2004

A change was made in the MPC control algorithm after snow event 3. New logic was added to the algorithm: if the current bridge surface temperature is below the desired temperature set point, the Bridge Loop supply temperature set point is automatically set to upper limit of heating system, which means max heat capacity is used.

The ambient air temperature for snow event 4, February 4, 2004, is shown in Figure 4-19. The air temperature was above 2°C before the snow event. Then it dropped to 0°C and remained around 0°C during the entire period.

According to the observation record, snow started around 10:30, February 4 and ended at about 18:00, February 4.

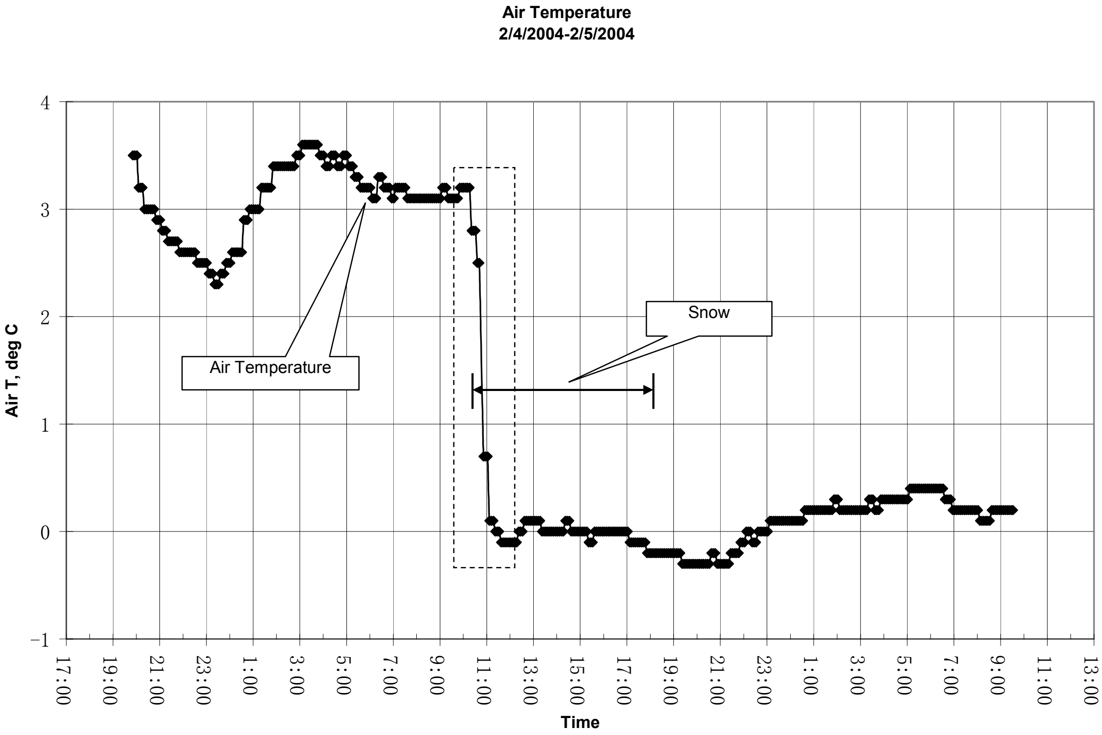


Figure 4-19: Event 4 Air Temperature

The controller parameters were set as follows: Dew Point Depression Threshold = 3°C; Controller Execution Period = 15 minutes; Bridge Deck Set Point Temperature = 4°C; Warm up Safety Period = 2 hours; Bridge Reference Heat Rate = 0.5°C/hr. The settings mean the bridge surface temperature should be heated to 4°C at least two hours before the predicted ice threat.

The controller first engaged the heating system at 01:36, nine hours before the snow began. After several on-off cycles it kept the heating system on from 10:52, 2/4/04 to 09:00, 2/5/04 (Figure 4-20). Early cycling was due to the same reason described for event 2.

.

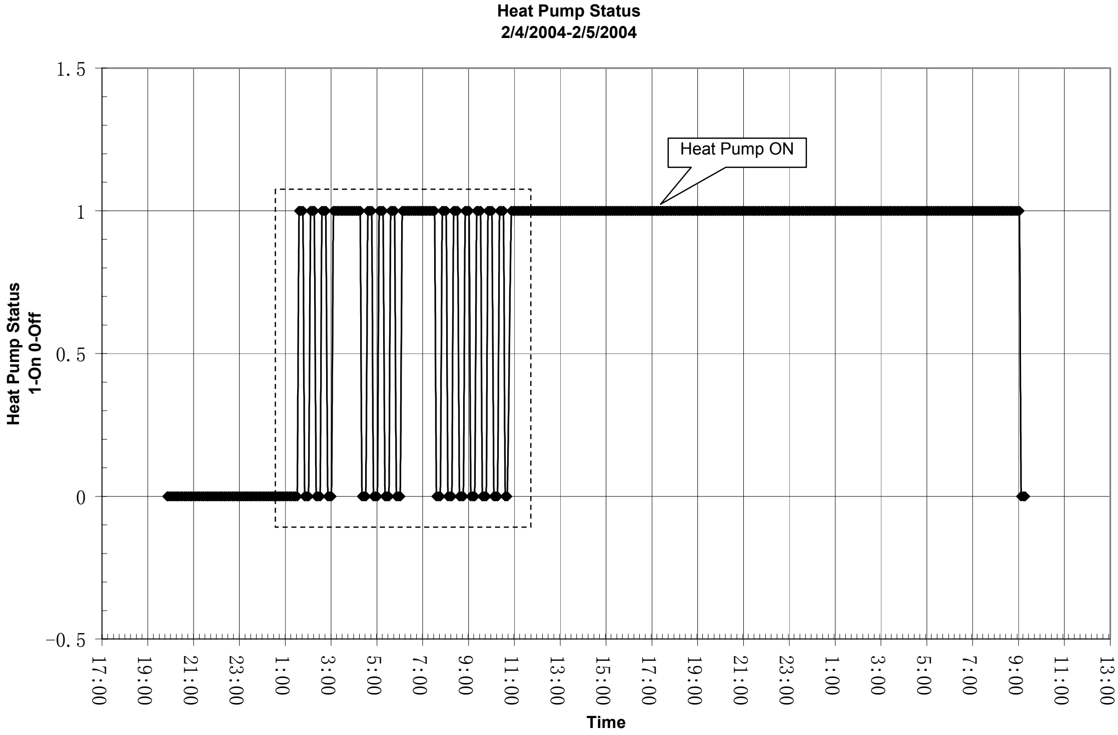


Figure 4-20: Event 4 Heat Pump Status

The air temperature dropped significantly in the first hour of the snow, as did the bridge surface temperature. Though the control algorithm raised the Bridge Loop temperature set point to 50°C, which is the upper limit of the heating system, the actual Bridge Loop supply temperature could only reach about 38°C. This confirms a conclusion from the Event 3 analysis, the heating system of the OSU test bridge is undersized.

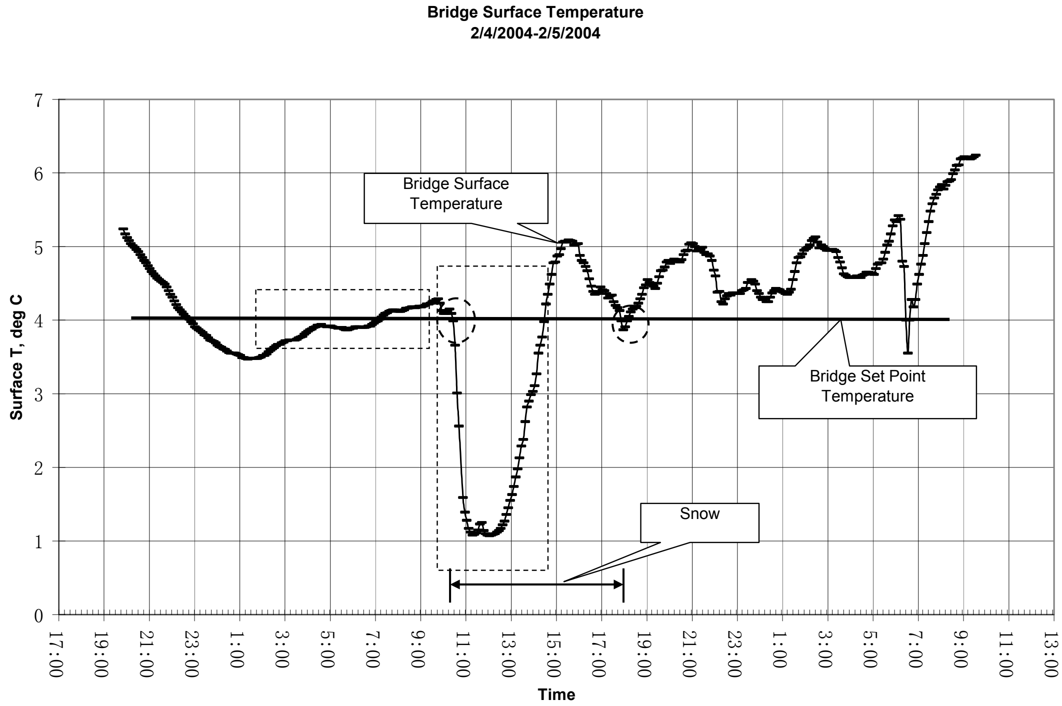


Figure 4-21: Event 4 Bridge Surface Temperature

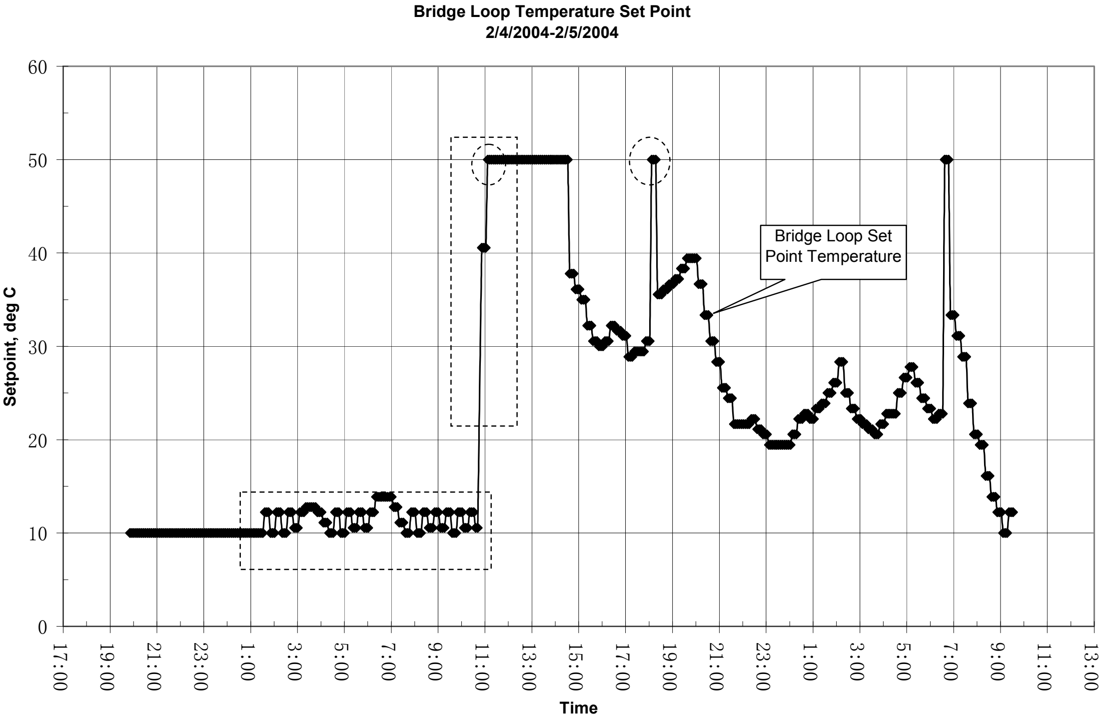


Figure 4-22: Event 4 Bridge Loop Temperature Set Point

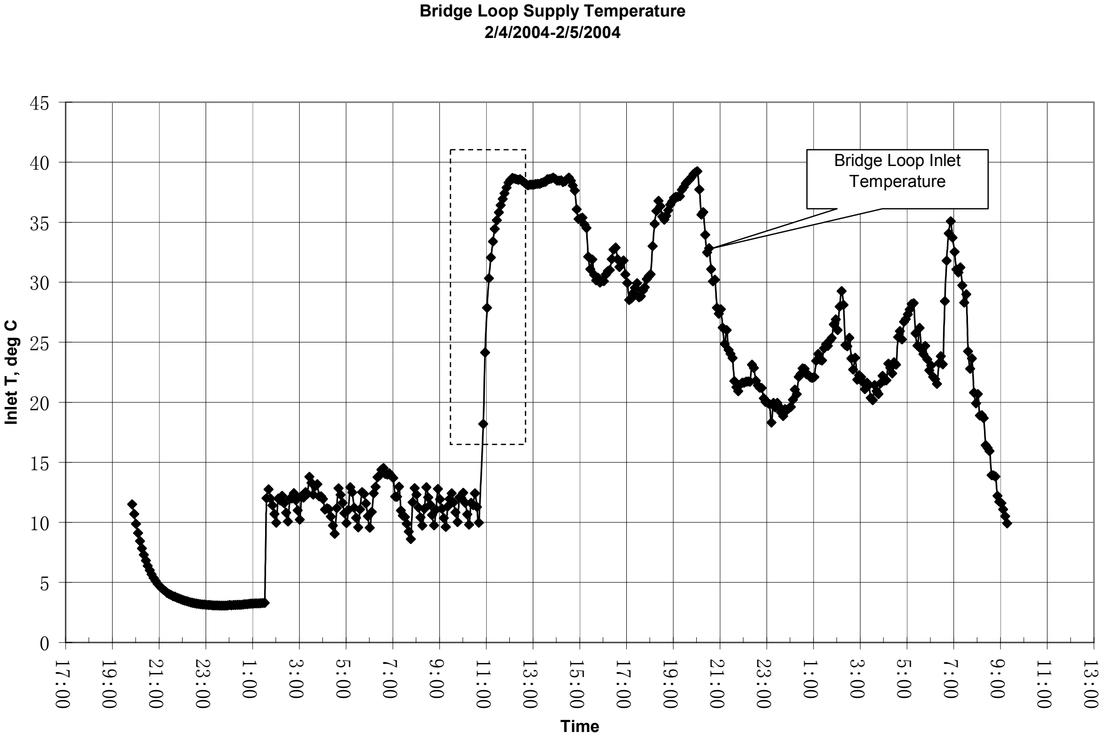


Figure 4-23: Event 4 Bridge Loop Supply Temperature

The overall performance of the control system was good in this snow event. Though the bridge surface temperature reached its lowest point at 11:30, there was still no snow accumulation on the heated part of the bridge (Figure 4-24). At 12:30, during the heaviest snow fall, the non-heated part of the bridge was covered by snow; while the heated part remained clear (Figure 4-25).

Conclusion: The control system detected the on-coming snow event and preheated the bridge in the proper manner. No snow accumulation on the bridge. The new logic added to the MPC algorithm improved performance.

Figure 4-24: Event 4 Bridge Snapshot at 11:30, 2/4/2004

(Snow, +1 hours)

Figure 4-25: Event 4 Bridge Snapshot at 12:30, 2/4/2004

(Snow, +2 hours)

4.6 Conclusion

The control system's performance during snow events 2 and 4 demonstrated the feasibility of the advance weather forecast and MPC algorithm. The controller can detect an on-coming ice event and preheat the bridge in the proper manner. Therefore, snow or ice accumulation on the bridge can be prevented. The operating costs can be kept at a minimum because the heating system is only used at times required.

The RUC weather forecast plays an important role in the controller. In snow events 1 and 3, the losses of forecast prevented controller from taking preheat action. The reliability of the RUC forecast feed needs to be improved.

The new logic made to the MPC algorithm improved the performance. However, large changes of the manipulated variable are not preferred from the view point of control engineer. The MPC algorithm needs additional tuning.

CHAPTER V

CASE STUDY PART II – SIMULATIONS

5.1 MPC Weight Specification Problem

In chapter IV, the real time performance of control system was analyzed. While the performance was generally satisfactory, a common problem was observed with the rate of bridge preheating. When the bridge surface temperature was below the set point of the beginning of a preheating period, the controller generated control action that caused the bridge temperature to follow the reference trajectory with a constant offset. This offset led to less preheat than desires when each snow event began.

As described in Chapter III, the MPC control action is obtained by solving an optimization problem:

$$\min_{\Delta u} \Phi = \left(\hat{e} - A\,\Delta u\right)\Gamma^{T}\Gamma\left(\hat{e} - A\,\Delta u\right) + \Delta u^{T}\Lambda^{T}\Lambda\,\Delta u \qquad \text{(Eq. 5-1)}$$

The right hand side of the objective function contains two parts: the error penalty term $\left(\hat{e} - A\,\Delta u\right)\Gamma^{T}\Gamma\left(\hat{e} - A\,\Delta u\right)$, and the move suppression term $\Delta u^{T}\Lambda^{T}\Lambda\,\Delta u$.

Large moves by the MV produce a quick system response and reduce the value of error penalty term. However, there is a corresponding increase in the move suppression term. This means the optimal result can only be obtained when the two terms reach some kind of balance. This can be easily understood by the following example.

Consider the situation in Figure 5-1. SP is the desired bridge temperature (a constant in this case) or reference trajectory. O is the initial actual bridge deck temperature. Consider the system response under two MV actions, u1 and u2. Case a shows the system response to MV action u1; case b, shows the system response to MV action u2. It is clear that response A is faster than B. In addition, the difference between A and SP (calculated by $\int (SP - A)^2 dt$) is smaller than the difference between B and SP ($\int (SP - B)^2 dt$). This means the unweighted error penalty term in case a is smaller than that in case b.

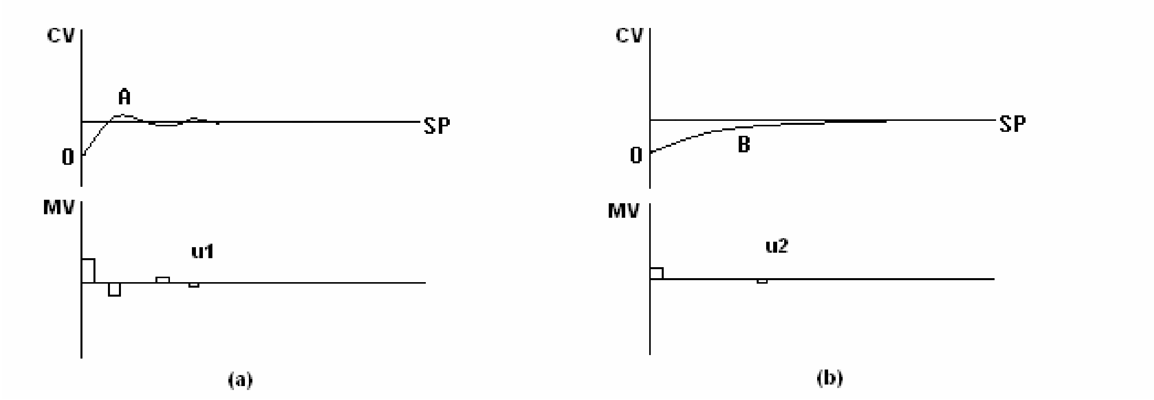


Figure 5-1: MV-CV Relations

However, to obtain a quick response, more control effort is needed in case a. This produces a larger unweighted suppression term for case a. Therefore, the value of objective function in case a may not be smaller than that in case b. The determining factor will be the weighting factors for the error penalty and move suppression terms. The control quality can be changed by adjusting the Γ and Λ matrices.

The preheat lag problem indicates the need for better value for Γ and Λ matrices. However, tuning the elements of these matrices is not easy. Very little guidance can be found. It is more an art than a science [26].

The remainder of this chapter presents case studies showing the response from different sets of matrix elements.

5.2 Case Study Basics

All the case studies were generated by using the simulation mode provided by the Smart Bridge control system software.

The simulations used the actual weather data recorded for the Stillwater Mesonet station from 18:06, 1/30/2004 to 10:26, 1/31/2004, which covered a typical winter night. The simulated RUC forecast was generated from the Mesonet data log. This condition means that the controller has a perfect forecast. For example, at time A, the Mesonet data at A+1 hour is used as the 1 hour forecast information. The only modification was the addition of a snow at 3:00, 1/31/2004.

The simulation controller parameters were: Dew Point Depression Threshold = 0°C; Controller Execution Period = 15 minutes; Bridge Deck Set Point Temperature = 2°C; Warm up Safety Period = 1.5 hour; Bridge Reference Heat Rate = 0.5°C/hr.

The Dew Point Depression Threshold of 0°C means that the dew point depression rule is turned off. The snow is the only indicator of an ice/snow event. This produces a fixed reference trajectory in all case studies which simplifies analysis of the results. As an additional simplification, the Bridge Loop supply temperature was assumed to track the Bridge Loop temperature set point perfectly in these case studies.

Figure 5-2 shows the air temperature during these simulations.

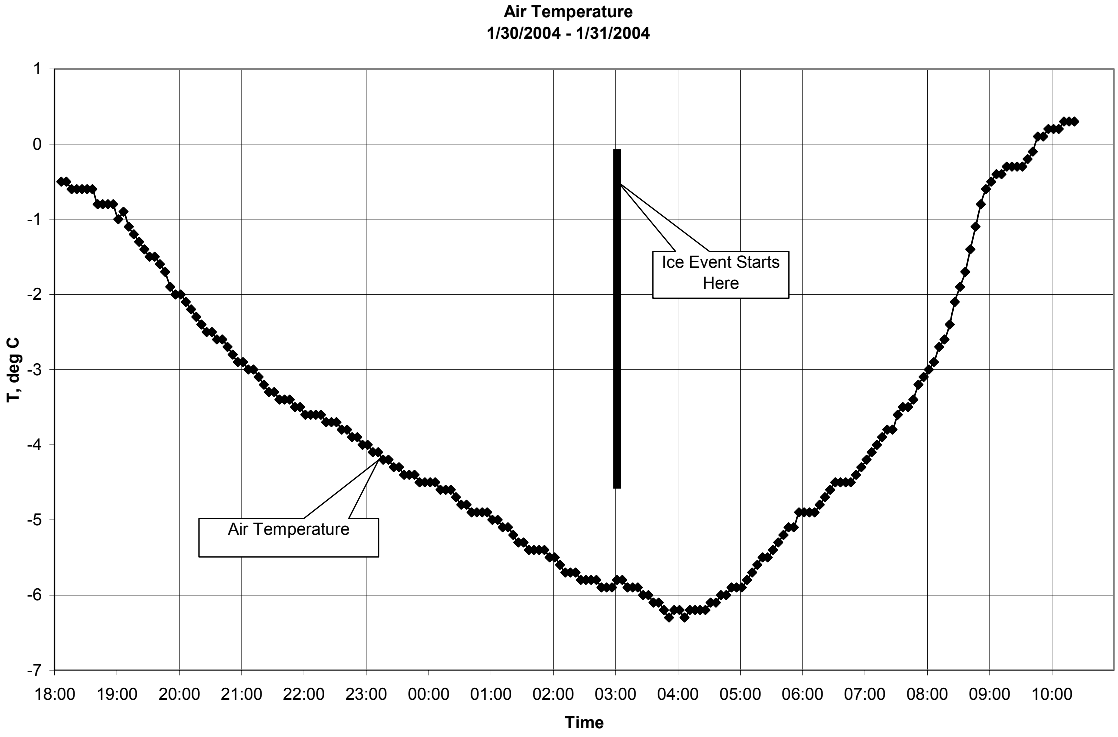


Figure 5-2: Air Temperature

5.3 Case Study 1

In this case study, the diagonal elements of the Γ (error penalty) matrix were set to: γ(1) ~ γ(16) = 2.5, γ(17) ~ γ(32) = 1.0, γ(33) ~ γ(48) = 0.5. The diagonal elements of the Λ (move suppression) matrix were set to: λ(1) ~ λ(24) = 1. These were the values recommended Jenks [13] and were used to produce the real-time results described in Chapter IV. The motivation for dividing Γ matrix into three parts is to force the control system to respond differently in the near, middle, and distant future. Jenks' choice of weights [13] puts the greatest weight on near term controller performance.

The controller turned on the heat pump at 18:26, 1/30/2004, about nine hours before the ice event (Figure 5-3) and kept the heat pump on until 9:00, 1/31/2004.

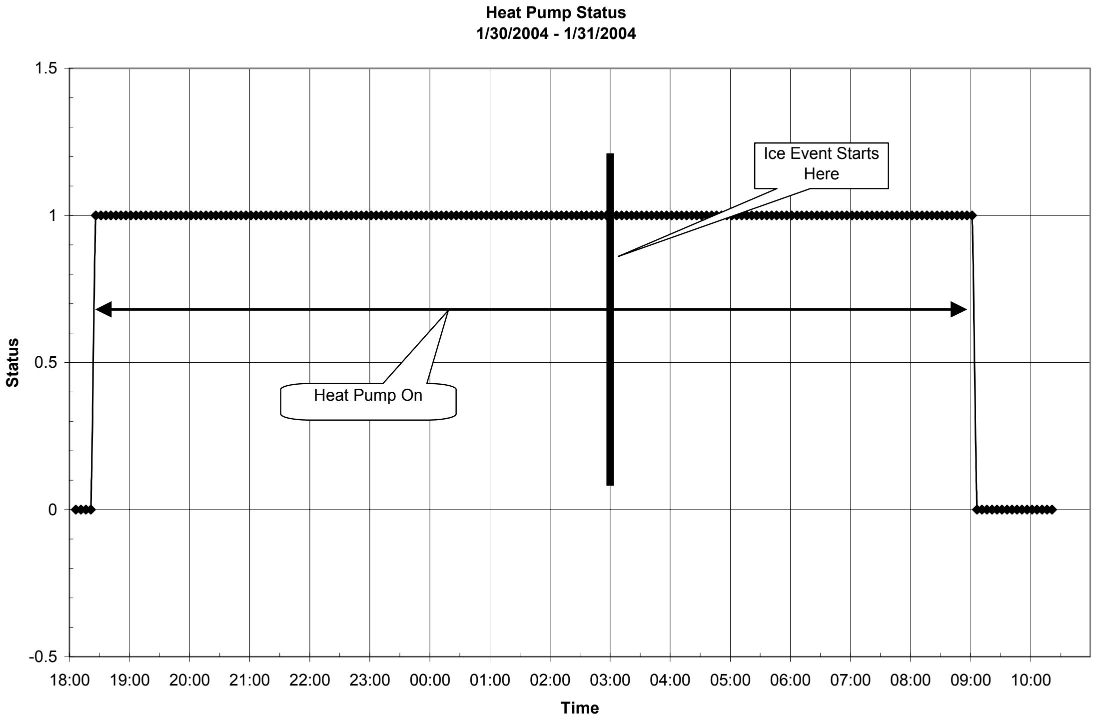


Figure 5-3: Heat Pump Status

The bridge surface temperature and the Bridge Loop set point trends are shown in Figures 5-4 and 5-5, respectively.

In Figure 5-4, a near-constant lag of one hour can be noticed between the trends of bridge surface temperature and the set point reference trajectory. The bridge temperature remained below the set point trajectory until the time of the simulated ice event occurred.

Because a perfect forecast was used in this case study, this result clearly indicated that the Γ and Λ matrices in MPC algorithm were not properly specified.

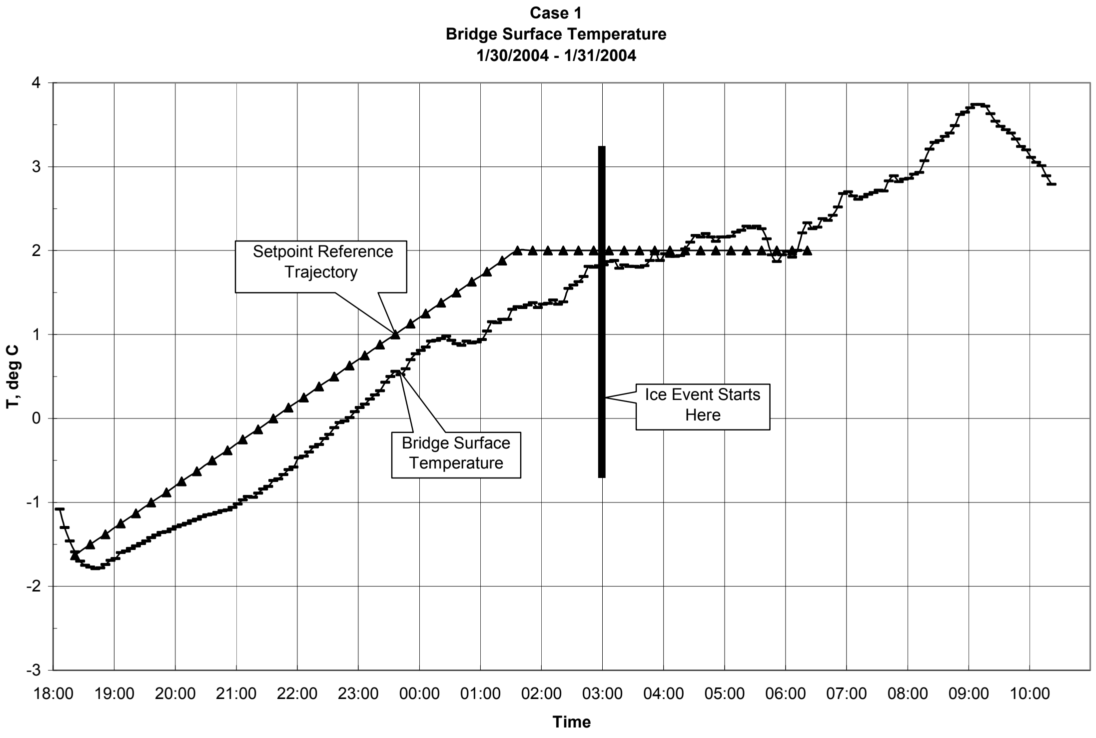


Figure 5-4: Case 1 Bridge Surface Temperature

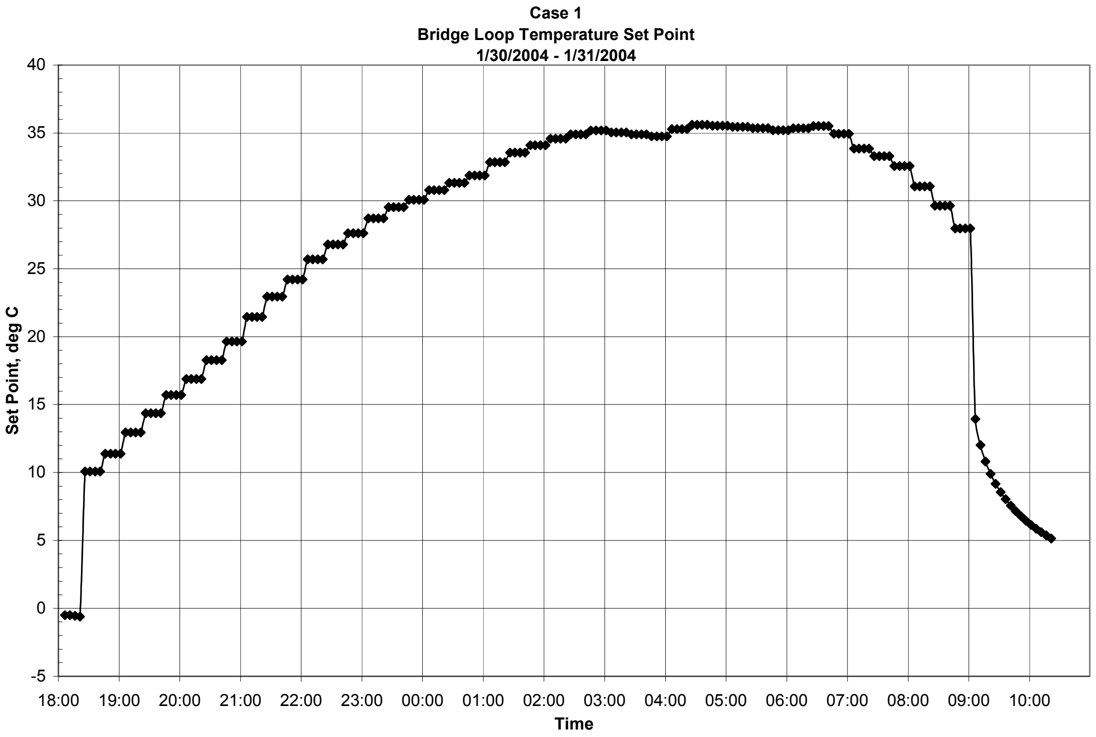


Figure 5-5: Case 1 Bridge Loop Temperature Set Point

5.4 Case Study 2

The previous case study showed the need to increase the Bridge Loop temperature more quickly. This means large control actions should be allowed. Because the controller only implements the first control action calculated by the MPC algorithm, more emphasis should be made on the control action in the near future. Therefore, in this case study, the Λ (move suppression) matrix was also divided into three parts, $\lambda(1) \sim \lambda(8)$, $\lambda(9) \sim \lambda(16)$ and $\lambda(17) \sim \lambda(24)$ in the same manner as the Γ (error penalty) matrix. Smaller values should be assigned to $\lambda(1) \sim \lambda(8)$ than $\lambda(9) \sim \lambda(16)$ and $\lambda(17) \sim \lambda(24)$. This allows more aggressive control action in the near future.

For this case study, the Γ (error penalty) matrix was modified as: $\gamma(1) \sim \gamma(16) = 3$, $\gamma(16) \sim \gamma(32) = 0.5$, and $\gamma(33) \sim \gamma(48) = 0.1$; the Λ (move suppression) matrix was set as: $\lambda(1) \sim \lambda(8) = 0.1$, $\lambda(9) \sim \lambda(16) = 0.5$ and $\lambda(17) \sim \lambda(24) = 1$. Compared to Case Study 1, these weights should emphasize near term error reduction with fewer penalties on aggressive control action.

The resulting bridge surface temperature trend is presented in Figure 5-6.

In this case, the bridge surface temperature tracked the set point reference trajectory very well and reached the desired 2°C set point about one hour before the ice/snow event.

Comparison of the Bridge Loop temperature set point trends for Case Study 1 and 2 is shown in Figure 5-7. The control actions for Case Study 2 were more aggressive and produced the desired system response. The lag noticed in Case Study 1 was eliminated.

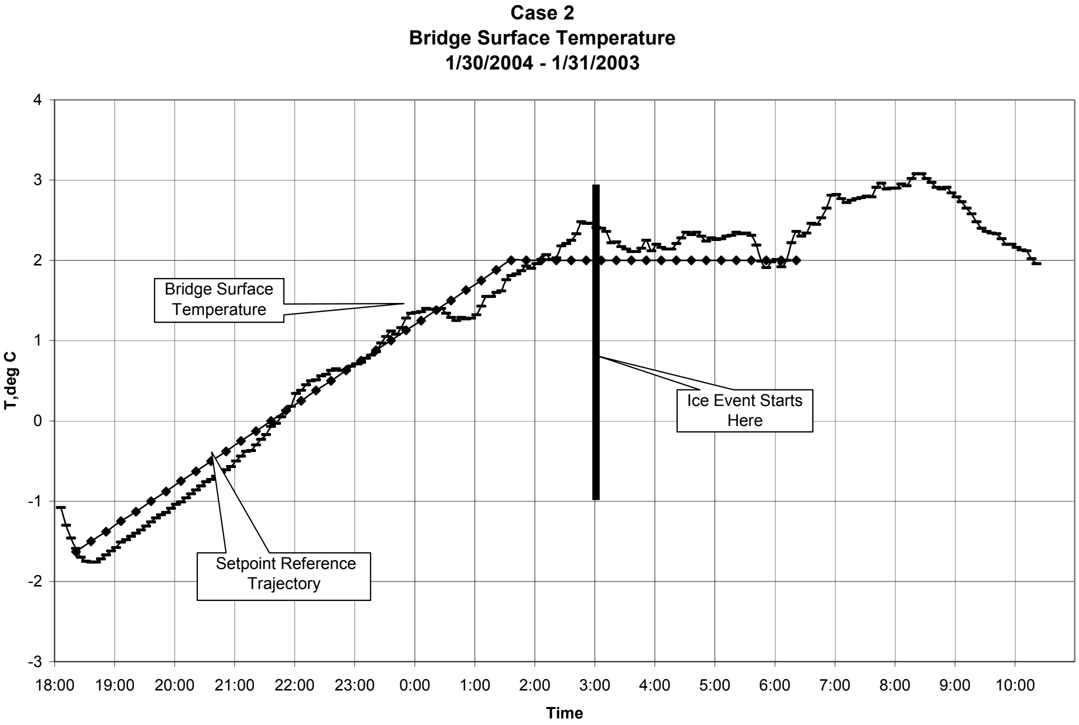


Figure 5-6: Case 2 Bridge Surface Temperature

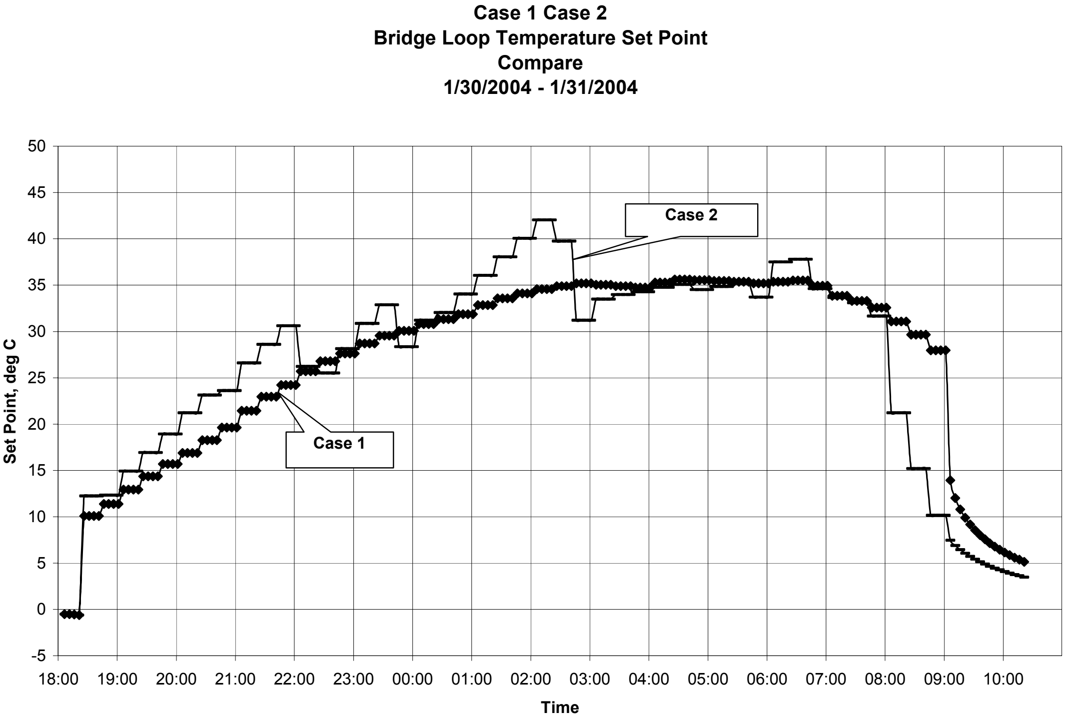


Figure 5-7: Bridge Loop Set Point Compare

5.5 Case Study 3

In this case study, the Γ (error penalty) matrix was set as: $\gamma(1) \sim \gamma(16) = 0.75$, $\gamma(16) \sim \gamma(32) = 1.25$, and $\gamma(33) \sim \gamma(48) = 1$. The change in Γ from Case Study 2 reduced the near-term penalty on tracking error with increased emphasis on mid and long-term error reduction. The Λ (move suppression) matrix was the same as in Case Study 2: $\lambda(1) \sim \lambda(8) = 0.1$, $\lambda(9) \sim \lambda(16) = 0.5$ and $\lambda(17) \sim \lambda(24) = 1$.

This setting of the matrices generated an interesting result. Figure 5-8 shows the bridge surface temperature trend. A constant lead of bridge surface temperature has now been produced. Figure 5-9 shows the Bridge Loop temperature set point trajectory.

When the heating system was engaged, the Bridge Loop set point temperature was raised very quickly. This accelerated the initial preheat rate relative to Case Study 2. The Bridge Loop temperature set point was then reduced before settling into a rate of increase similar to Case Study 2.

The bridge surface climbed almost parallel to reference trajectory and reached the desired bridge temperate 2ºC three hours before the ice event. This has the benefit of storing more heat in the bridge prior to onset of the snow event.

The bridge temperature was maintained within 0.5ºC of the desired bridge temperature throughout the ice event.

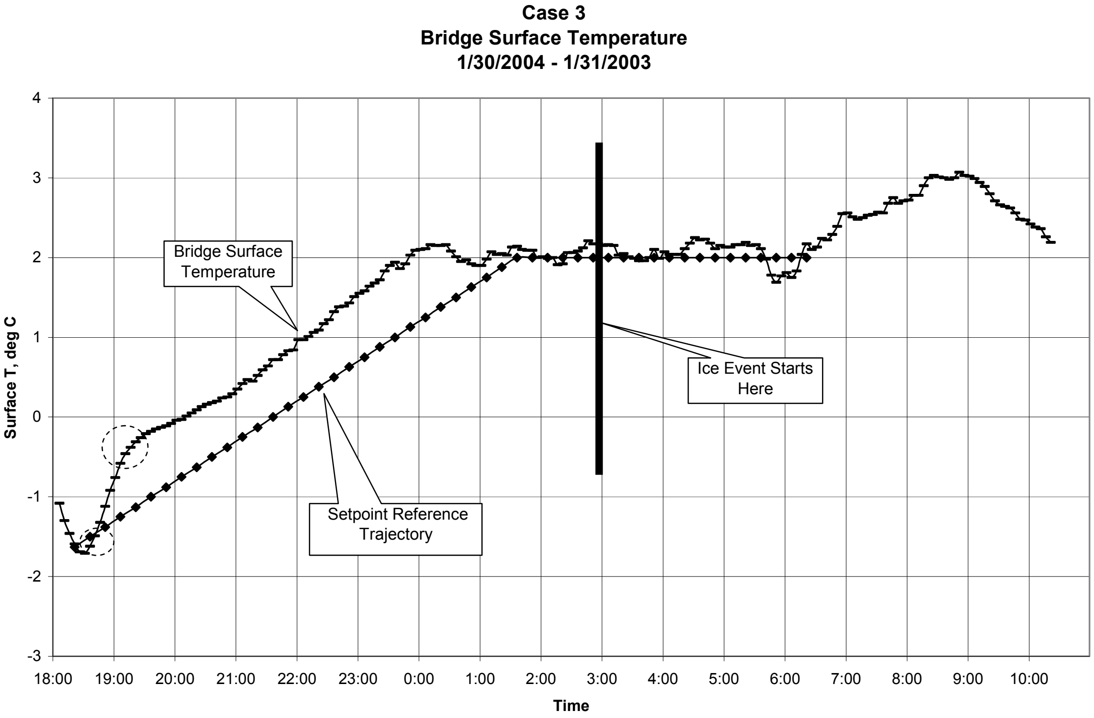


Figure 5-8: Case 3 Bridge Surface Temperature

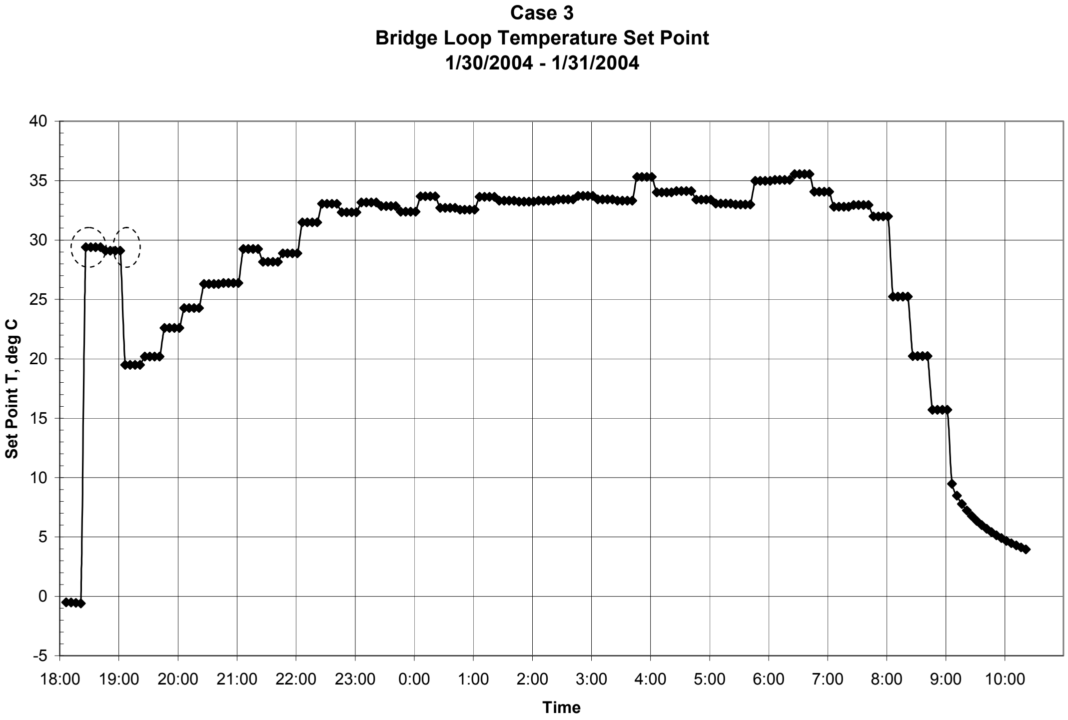


Figure 5-9: Case 3 Bridge Loop Set Point

5.6 Conclusion

The case studies in this chapter confirmed that the MPC controller’s performance could be improved by tuning the elements of the Γ and Λ matrices. By dividing the matrices into several sections and changing them separately, better controller performance results were obtained. This method can be a starting point to find appropriate weight matrices.

The matrices used in case studies can be used in the future real time testing.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

This project shows the successful application of model prediction control (MPC) technology for “smart control” of a geothermal heated bridge. By combining MPC with digital forecast information provided by the National Weather Service, the control system proved capable of maintaining ice-free bridge deck conditions within the limits of the bridge heating system. Conclusions from this demonstration work follow.

First, the work reported in this thesis represents a major advance in heated bridge technology. The control systems used in previous heated bridge projects utilized On/Off feedback techniques with local weather information. The control system developed in this thesis utilizes advanced weather forecast and model predictive control techniques. The advantage of using this control system is that the heating system is engaged before icing conditions reach the bridge and the bridge temperature is controlled with optimized heat input. As a result, the bridge can be pre-heated to a desired temperature before the icing events, which guarantees the ice-free bridge condition.

Second, this work demonstrates the ability to access and leverage real-time weather forecast data. By using the RUC weather forecast and appropriate icing event detection rules, the controller is capable of identifying approaching icing events. This recognition is critical to give the heating system enough time to pre-heat the bridge.

Third, this work documents a successful application of model predictive control technology in a novel application. By using MPC, the bridge temperature can be controlled to follow an optimized reference trajectory. This optimization minimizes the demands on the heating system with a corresponding minimization of the heating system capital and operation costs.

Fourth, the demonstrated control system has been implemented using robust, user friendly, modular and object oriented control system software. The control system software uses a variety of methods to overcome real-time operating issues not encountered in the original controller software [13]. The modular and objective oriented features of the new controller software facilitate future improvements and technical transfer.

Fifth, this work identified MPC controller tuning adjustments that can provide improved control performance. The results can be used as a starting point for future work to develop generalized tuning guidelines.

6.2 Recommendations and Future Work

The most serious problems encountered were the loss of the RUC weather forecasts and the variability in timeliness of the Mesonet weather information. During the first and third snow events (Chapter IV), the RUC forecasts were not available prior to the snow event. Consequently, the controller had no warning of the approaching snow and was unable to preheat the bridge. This means the reliability of the RUC weather forecast feed needs to be improved. Although outside the scope of the control engineer, improvement in this area represents the greatest need at present.

Weather conditions were observed to change dramatically at the beginning of each snow event. Unfortunately, Mesonet information was typically received thirty to forty minutes after being issued. This lag affects the performance of the controller during the critical initial part of a snow event. This problem points out the need to provide local weather measurement capability at the bridge site.

LIST OF REFERENCES

[1] Benjamin, S. G., J. M. Brown, K. J. Brundage, B. E. Schwartz, T. G. Smirnova, and T. L. Smith, "*RUC-2: The Rapid Update Cycle - Version 2*," http://maps.fsl.noaa.gov/ruc2.tpb.html, 1998

[2] Benjamin, S. G., "*Present and Future of the Rapid Update Cycle*," http://maps.fsl.noaa.gov/CWSU/ruc_feb00_CWSU_files/v3_document.htm, 2000

[3] Callihan, B. K., "*A Control System Framework for a Bridge Deck Heated by a Geothermal Heat Pump System*," M.S. Thesis, Oklahoma State University, Stillwater, OK, 2000

[4] Chiasson, A., "*Advances in Modeling of Ground-Source Heat Pump Systems*," M.S. Thesis, Oklahoma State University, Stillwater, OK, 1999

[5] Cutler, C. R. and Ramaker, B. L., "*Dynamic Matrix Control – A Computer Control Algorithm*," The National Meeting of AICHE, Houston, TX, 1979

[6] Derek S. Arndt, darndt@ou.edu, Assistant Oklahoma State Climatologist, Oklahoma Climatological Survey, Norman, OK

[7] Dunbar, W. B., "*Distributed Receding Horizon Control of Multiagent System*," Ph.D. Thesis, California Institute of Technology, Pasadena, California, 2004

[8] "*Fatality Analysis Reporting System, 2003*," National Highway Traffic Safety Administration

[9] Fletcher, R., "*Practical Methods of Optimization*," John Wiley & Sons Ltd. 1987

[10] Garcia, C. E. and Morshedi, A. M., "*Quadratic Programming Solution of Dynamic Matrix Control (QDMC)*," Chem. Eng. Commun. Vol. 46 pp. 073-087

[11] High, K. A., "*Optimization Applications Course Notes*," CHE 5703, Oklahoma State University, 2003

[12] Hornbeck, R. W., "*Numerical Methods*," Quantum Publishers, Inc, 1975

[13] Jenks, S. C., "*A Model Predictive Control Strategy for a Bridge Deck Heated by a Geothermal Heat Pump System*," M.S. Thesis, Oklahoma State University, Stillwater, OK, 2001

[14] "*Manual of Practice for an Effective Anti-Icing Program*," US Army Cold Regions Research and Engineering Laboratory

[15] Marlin, T. E., "*Process Control*," McGraw Hill, 1995

[16] Mayne, D.Q. etc., "*Constrained Model Predictive Control: Stability and Optimality*," Automatica, 36 p789-814, 2000

[17] Minsk, L. D., "*Heated Bridge Technology: Report on ISTEA Sec. 6005 Program*," Publication No. FHWA-RD-99-158, U.S. Department of Transportation, Federal Highway Administration, Office of Bridge Technology, 1999

[18] Morari, M. etc., "*Model Predictive Control: Past Present and Future*," Computers and Chemical Engineering, 23 p667-682, 1999

[19] Ogunnaike, B. A. and Ray, W. H. "*Process Dynamics, Modeling, and Control*," Oxford, 1994

[20] "*Oklahoma State University Geothermal Smart Bridge Proposal*," Oklahoma State University, Stillwater, OK, 1999

[21] Ramamoorthy, M., "*Applications of Hybrid Ground Source Heat Pump Systems to Buildings and bridge Decks*," Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK, 2001

[22] Rawlings, "*Model Predictive Control technology, Theory and Applications DMC and QDMC*," Short Course Notes, 1997

[23] Spitler, J. D. and M. Ramamoorthy, "*Bridge Deck Deicing Using Geothermal Heat Pumps*," Proceedings of Heat Pumps in Cold Climates IV International Conference, Alymer, Quebec, 2000

[24] Spitler, J. D., "*Oklahoma State University Geothermal Smart Bridge*," http://www.smartbridge.okstate.edu, 2004

[25] "*The Oklahoma Mesonet – General Overview*," Oklahoma Mesonet, 2004

[26] Whiteley, J. R., "*Advanced Process Control Course Notes*," CHE 5853, Oklahoma State University, 2004

APPENDIX A: OBJECTS

TBridgeModel

Description: This object encapsulates the first principle bridge deck model. User can use this object to predict the bridge response in the future. Bridge parameters are stored inside this object.

Methods:

- ***GetBridgePara(aryBridgePara() As Double) As Boolean***

  Copy the bridge parameters to the output array aryBridgePara.

- ***SetBridgePara(aryBridgePara() As Double) As Boolean***

  Update the bridge parameters with the input array aryBridgePara.

- ***Initialize()***

  Object initialization

- ***LoadPara()***

  Load the bridge parameters from bridge parameter configuration file.

- ***SimBridgePredict(dteCurTime As Date, aryRUCData() As Double, dblSimInletTemp As Double, dblSimInletFlow As Double, dblCurrentBridgeT As Double) As Double()***

  Predict bridge response. dteCurTime is current time, aryRUCData is the RUC forecast, dblSimInletTemp is the current bridge loop supply temperature, dblSimInletFlow is the current bridge loop supply flow rate, dblCurrentBridgeT is the current bridge surface temperature. This function returns an array whose content is the bridge response in the future.

TCommunication

Description: This object encapsulates the communication between the master controller and the slave controller. User can send command to the slave controller and get bridge information from the slave controller.

Methodes:

- ***Initialize()***

  Object initialization.

- ***TalkToSlave() As Boolean***

  Execute one communication with slave controller.

- ***GetBridgeMeasurements(aryBridgeInfo() As Double)***

  Update the bridge measurement information to the output array aryBridgeInfo.

TController

Description: This object encapsulates controller. It carries the ice threat detection, trajectory generation, and control action calculation.

Methodes:

- ***Initialize()***

  Object initialization.

- ***GetCtlPara(aryCtlPara() As Double) As Boolean***

  Copy controller parameters to the output array aryCtlPara.

- ***SetCtlPara(aryCtlPara() As Double) As Boolean***

  Update controller parameters by the input array aryCtlPara.

- ***LoadPara()***

Load the control parameters from the controller parameter configuration file.

- ***Run()***

  Run the controller in different mode (Real time of Simulation)

- ***ICEThreatDetection(aryRUCData() As Double, aryPredBridgeData() As Double, arySetPointTrajectory() As Double) As Boolean***

  Detect the Ice threat based on the RUC forecast in aryRUCData and predicted bridge response in aryPredBridgeData. If threat detected, a reference trajectory will be built and output by arySetPointTrajectory.

- ***BuildSPTraj(aryRUCData() As Double, aryPredBridgeData() As Double, intThreatStartPoint As Integer) As Double()***

  Generate reference trajectory.

- ***QuadOptimizer(aryPredBridgeData() As Double, arySetPointTraj() As Double, blnUseSimForm As Boolean) As Double()***

  Calculate the control action.

TDisplay

Description: This object encapsulates the display issue. It output data to the screen.

Methodes:

- ***ShowRealMesonet(dteCurrent As Date, aryMesonetData() As Double)***

  Display the Mesonet weather information.

- ***ShowRealRUC(aryRUCData() As Double)***

  Display the RUC weather forecast information.

- ***ShowRealBridge(dteCurrent As Date, aryBridgeData() As Double)***

Display the bridge information.

- ***ShowRealPredBridge(dteCurrentTime As Date, aryPredBridgeData() As Double)***

  Display the predicted bridge response.

- ***ShowRealSetPointTrag(dteCurrentTime As Date, arySetPointTrajectory() As Double, blnClear As Boolean)***

  Display the set point reference trajectory.

TFile

Description: This object encapsulates the file operations.

Methods:

- ***NewFile(ByVal strFilePathName As String)***

  Generate a new file with the specified file name and location.

- ***AddLine(ByVal strFilePathName As String, ByVal strLineInfo As String)***

  Add a new line to an existing file.

- ***ReadLineAt(ByRef strLineInfo As String, ByVal strFilePathName As String, ByVal intLineNumber As Integer) As Boolean***

  Read a line at a specified line number from a file.

- ***DoesFileExist(ByVal strFilePathName As String) As Boolean***

  Check the existence of a file.

TLog

Description: This object encapsulates the log system.

Methods:

- ***LogProjStart(dteCurrent As Date, strLineInfo As String, dteStart As Date)***

Log the project starting time. This function is called when a project is opened.

- ***LogProjEnd(dteCurrent As Date, strLineInfo, dteEnd As Date)***

  Log the project ending time. This function is called when a project is closed.

- ***LogMesonet(strPath As String, dteTime As Date, aryMesonetData() As Double)***

  Log the Mesonet data. This function is called every five minutes.

- ***LogRUC(strPath As String, dteTime As Date, aryRUCData() As Double)***

  Log the RUC data. This function is called every fifteen minutes.

- ***LogController(strPath As String, dteTime As Date, aryBridgeInfo() As Double, strMode As String, blnPump As Boolean, dblSetpoint As Double)***

  Log the controller information. This function is called every five minutes.

- ***LogBridgeModel(strPath As String, dteTime As Date, aryPredBridge() As Double, arySetPointTraj() As Double)***

  Log the bridge model prediction and the set point trajectory. This function is called every fifteen minutes.

TMesonet

Description: This object encapsulates the operation of Mesonet data file.

Methods:

- ***Initialize()***

  Object initialization

- ***UpdateMes(dteCurrent As Date, aryResult() As Double) As Boolean***

  Generate the current weather information array. dteCurrent is the current time. Result is return in the aryResult.

TRUC

Description: This object encapsulates operation of RUC data file.

Methods:

- ***Initialize()***

  Object initialization

- ***Forecast(aryCurWeather() As Double, dteCurTime As Date, dteCurGMT As Date, aryResult() As Double) As Boolean***

  Generate the RUC weather forecast array. dteCurrent and dteCurGMT are the current time and current GMT time. Current weather information is input by aryCurWeather. Result is return in the aryResult.

TStepModel

Description: This object encapsulates step model generation.

Methods:

- ***Initialize()***

  Object initialization

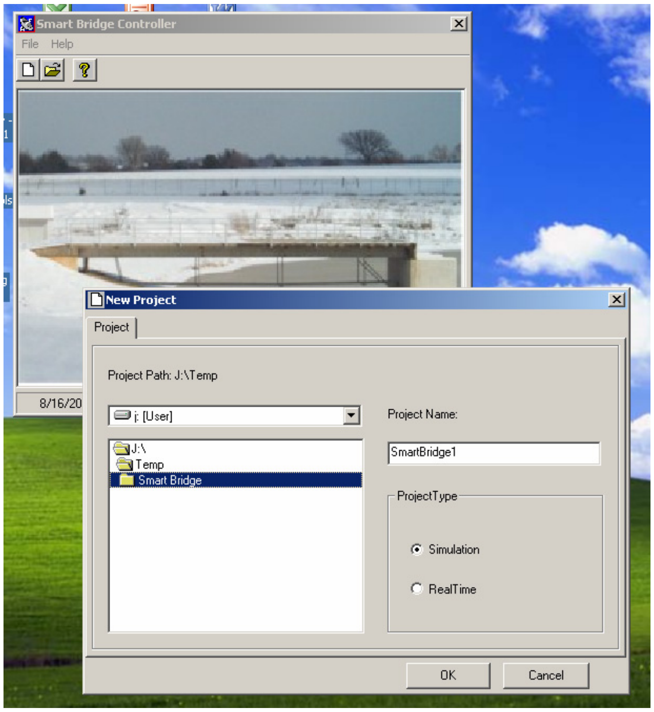
- ***GenerateStepModel()***

  Generate the step model.

APPENDIX B: PROJECT

A project is a structured collection of files containing bridge parameters, controller configuration information, weather measurements, weather forecasts, and historian files. With project, user can simulate controller performance or keep real time running results for different bridges.

When the Smart Bridge controller software is open, the user needs to choose either create a new project or open an existing project. The user can choose the directory, project name and project type for a new project (Figure B-1).



Project Path: J:\Temp

j: [User]

J:\
Temp
Smart Bridge

Project Name: SmartBridge1

ProjectType

● Simulation
○ RealTime

Figure B-1: New Project

After the user click ‘OK’, several files will be created in the user specified directory with the user chosen name (Figure B-2).

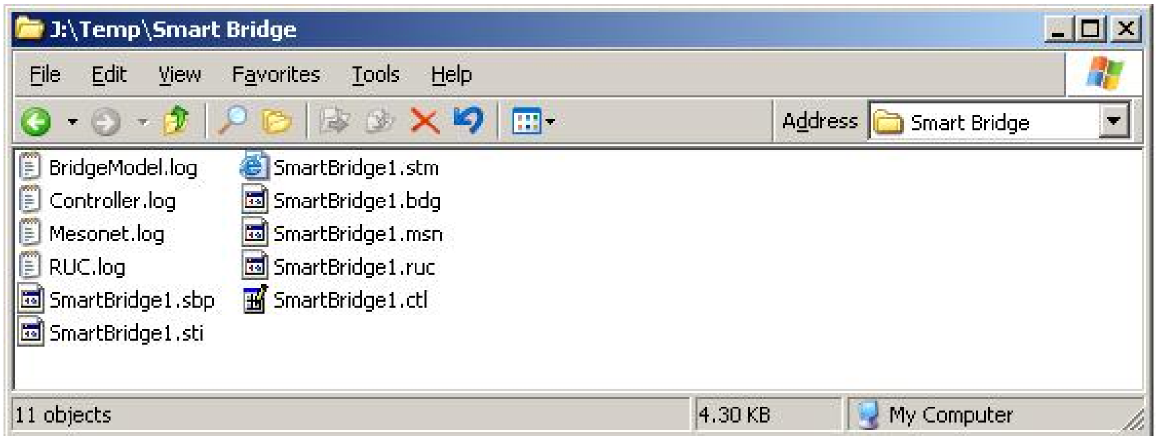


Figure B-2: Project Files

The ‘SmartBridge1.sbp’ file records the project type and the project’s running start time and end time.

The ‘SmartBridge1.sti’ records the step response initial weather conditions.

The ‘SmartBridge1.stm’ records the generated step response model.

The ‘SmartBridge1.bgd’ file records the bridge parameters.

The ‘SmartBridge1.msn’ file records the Mesonet data directory.

The ‘SmartBridge1.ruc’ file records the RUC data directory.

The ‘SmartBridge1.ctl’ file records the controller parameters.

A new entry will be added to the corresponding files when these configurations are changed.

The ‘BridgeModel.log’ file is the log file for bridge model prediction and set point trajectory. When the controller is running, a new entry is added to this file when an ice threat is detected.

The ‘Mesonet.log’ file is the log file for Mesonet data. When the controller is running, a new entry is added to this file every five minutes.

The ‘RUC.log’ file is the log file for RUC data. When the controller is running, a new entry is added to this file every fifteen minutes.

The ‘Controller.log’ file is the log file for controller variables. When the controller is running, a new entry is added to this file every five minutes.

APPENDIX C: INTERFACES

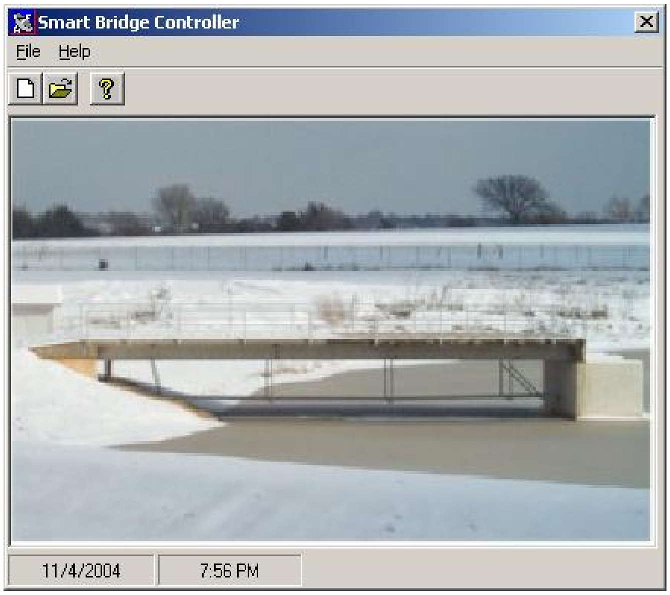
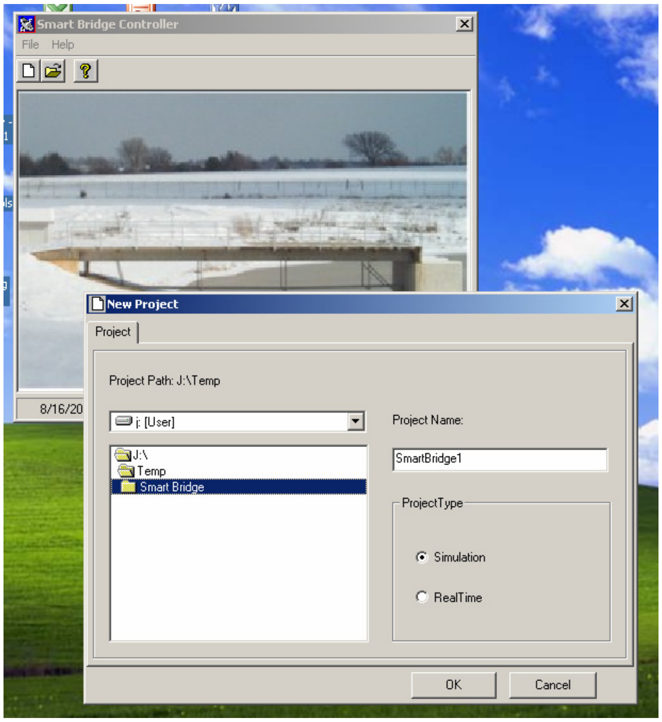


Figure C-1: Controller Welcome Window



Project Path: J:\Temp

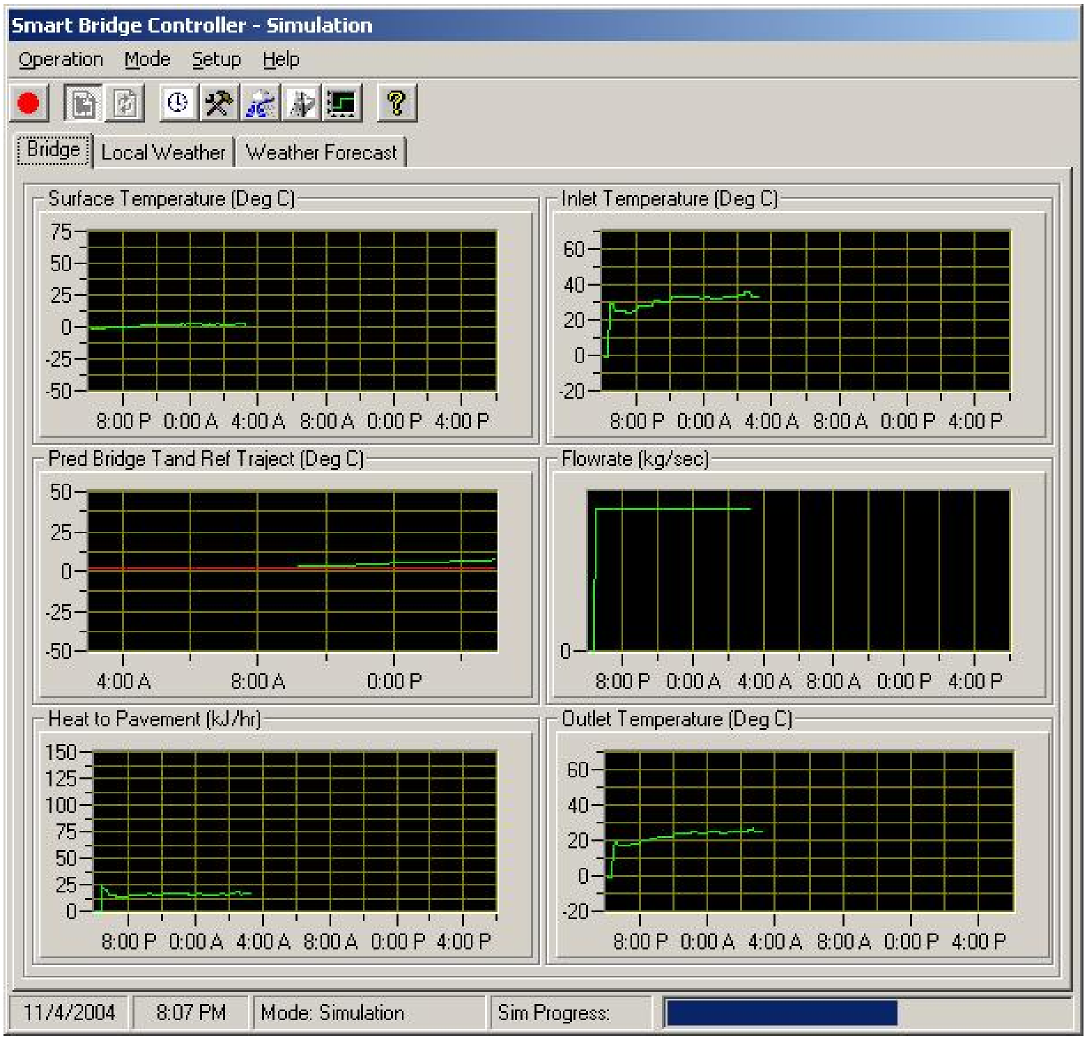
Project Name: SmartBridge1

ProjectType

● Simulation

○ RealTime

Figure C-2: New Project



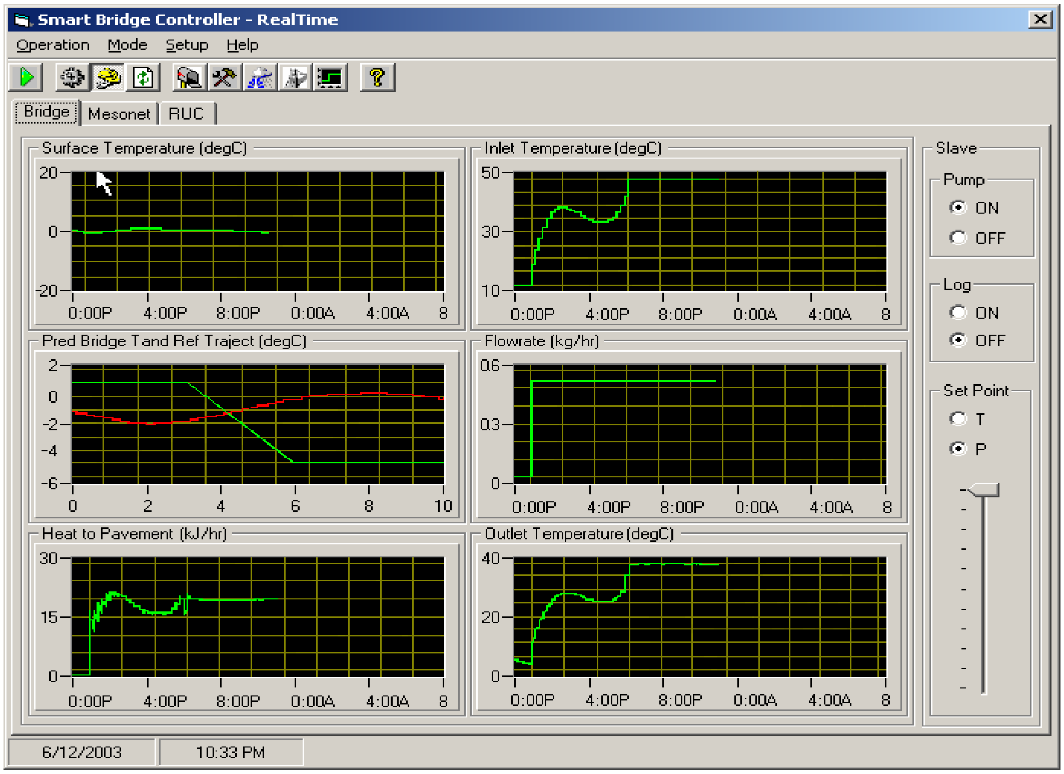
11/4/2004

8:07 PM

Mode: Simulation

Sim Progress:

Figure C-3: Simulation Interface



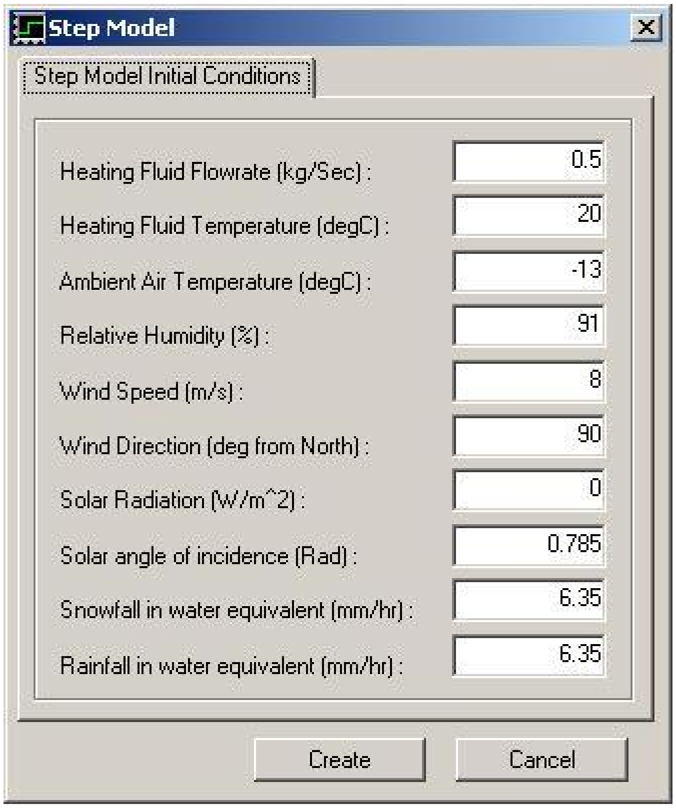
Pump
● ON
○ OFF

Log
○ ON
● OFF

Set Point
○ T
● P

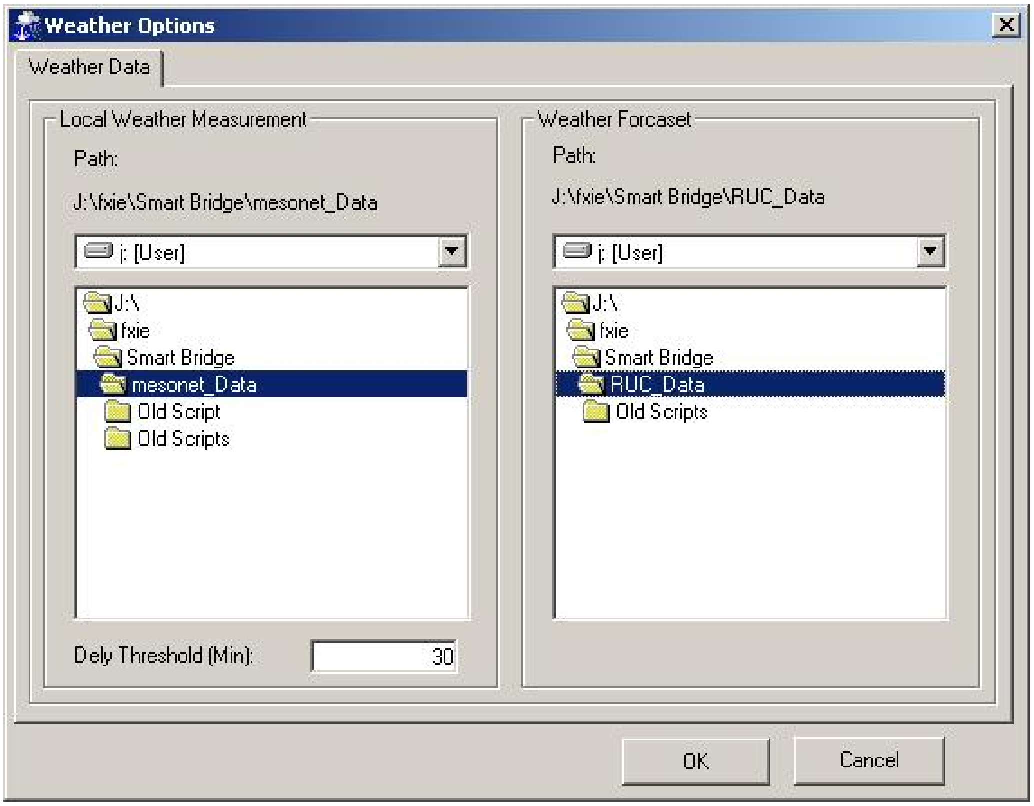
6/12/2003
10:33 PM

Figure C-4: Real Time Interface



Heating Fluid Flowrate (kg/Sec) :	0.5
Heating Fluid Temperature (degC) :	20
Ambient Air Temperature (degC) :	-13
Relative Humidity (%) :	91
Wind Speed (m/s) :	8
Wind Direction (deg from North) :	90
Solar Radiation (W/m^2) :	0
Solar angle of incidence (Rad) :	0.785
Snowfall in water equivalent (mm/hr) :	6.35
Rainfall in water equivalent (mm/hr) :	6.35

Figure C-5: Step Response Dialog

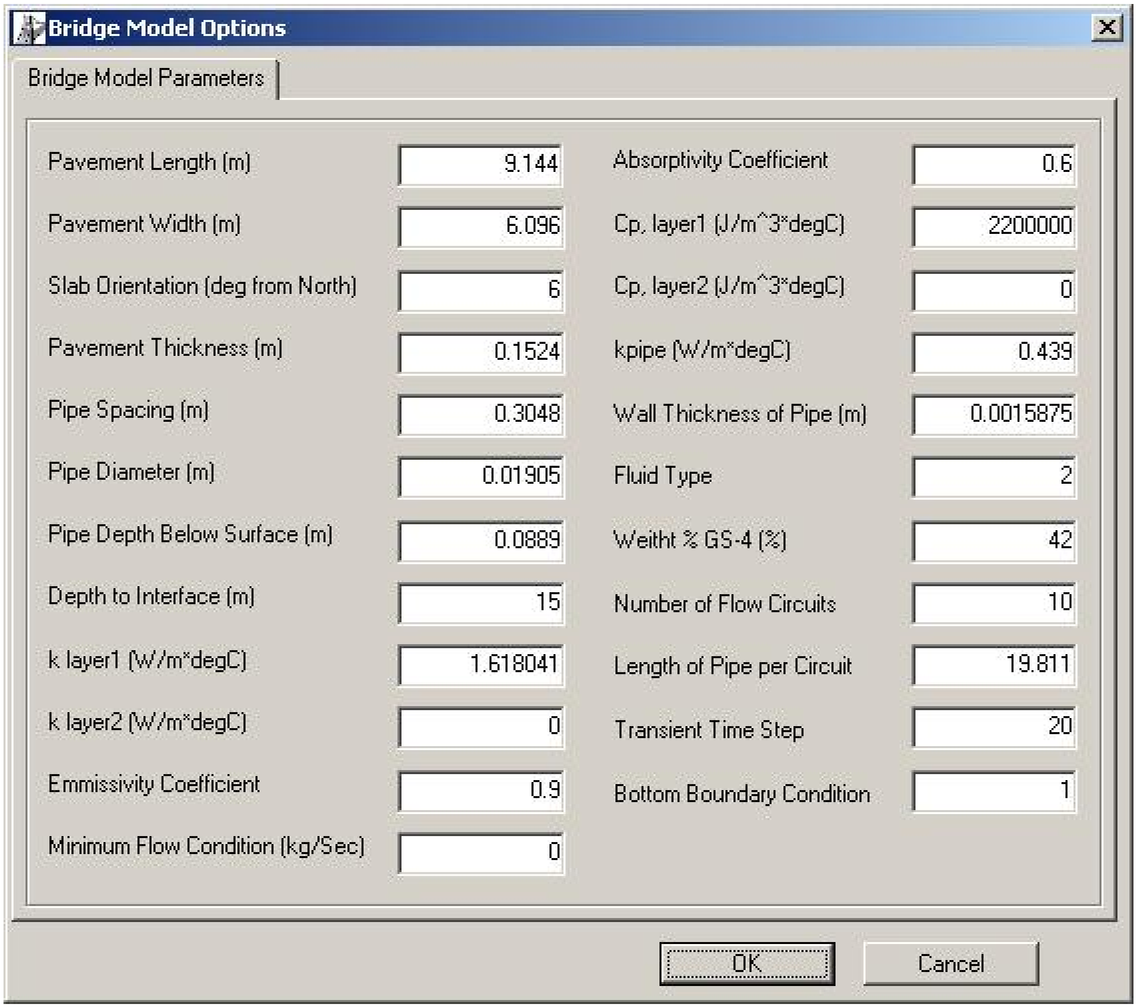


Path:
J:\fxie\Smart Bridge\mesonet_Data

Dely Threshold (Min): 30

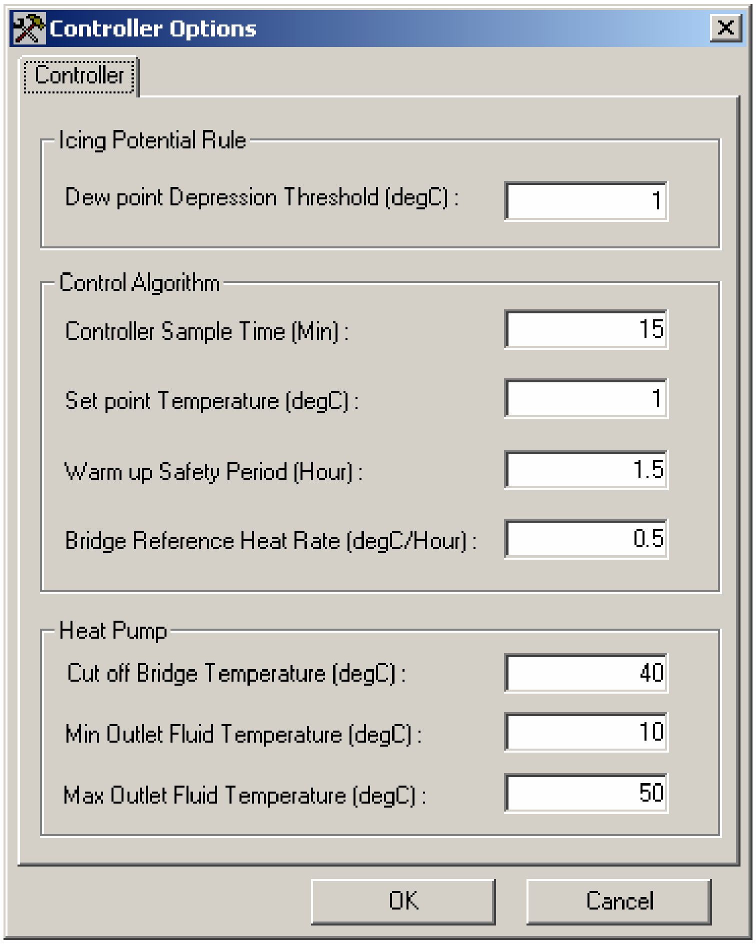
Path:
J:\fxie\Smart Bridge\RUC_Data

Figure C-6: Weather Input Dialog



Pavement Length (m)	9.144	Absorptivity Coefficient	0.6
Pavement Width (m)	6.096	Cp, layer1 (J/m^3*degC)	2200000
Slab Orientation (deg from North)	6	Cp, layer2 (J/m^3*degC)	0
Pavement Thickness (m)	0.1524	kpipe (W/m*degC)	0.439
Pipe Spacing (m)	0.3048	Wall Thickness of Pipe (m)	0.0015875
Pipe Diameter (m)	0.01905	Fluid Type	2
Pipe Depth Below Surface (m)	0.0889	Weitht % GS-4 (%)	42
Depth to Interface (m)	15	Number of Flow Circuits	10
k layer1 (W/m*degC)	1.618041	Length of Pipe per Circuit	19.811
k layer2 (W/m*degC)	0	Transient Time Step	20
Emmissivity Coefficient	0.9	Bottom Boundary Condition	1
Minimum Flow Condition (kg/Sec)	0		

Figure C-7: Bridge Parameters Dialog



Dew point Depression Threshold (degC) : 1

Controller Sample Time (Min) : 15

Set point Temperature (degC) : 1

Warm up Safety Period (Hour) : 1.5

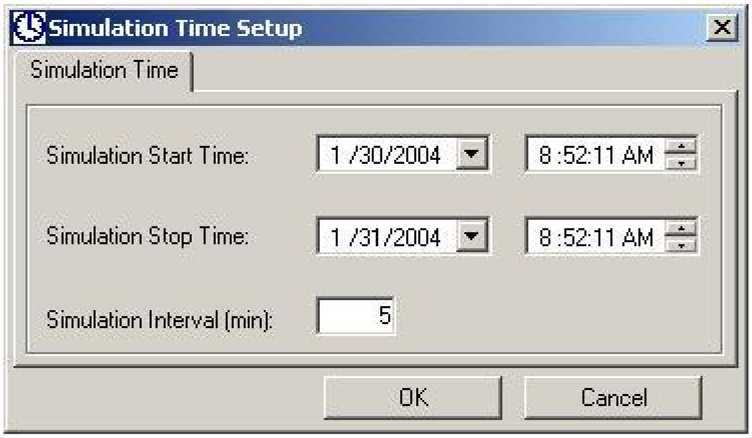
Bridge Reference Heat Rate (degC/Hour) : 0.5

Cut off Bridge Temperature (degC) : 40

Min Outlet Fluid Temperature (degC) : 10

Max Outlet Fluid Temperature (degC) : 50

Figure C-8: Controller Parameters Dialog



Simulation Start Time: 1 /30/2004 8 :52:11 AM

Simulation Stop Time: 1 /31/2004 8 :52:11 AM

Simulation Interval (min): 5

Figure C-9: Simulation Time Dialog

VITA

Feng Xie

Candidate for the Degree of

Master of Science

Thesis: APPLICATION OF MODEL PREDICTIVE CONTROL
TO A GEOTHERMAL HEATED BRDIGE DECK

Major Field: Chemical Engineering

Biographical:

Education:

Graduated from Hangzhou No. 15 High School, Hangzhou, China, in June 1993; Received Bachelor of Science Degree in Testing Technology and Instrumentation from Zhejiang University in June 1997; Received Master of Science Degree in Control Science and Engineering from Zhejiang University in March 2000; Completed the requirements for Master of Science degree in Chemical Engineering at Oklahoma State University in December, 2004.

Professional Experience:

Research Assistant, Department of Chemical Engineering, Oklahoma State University, August 2002 to July 2004.

Assistant lecture Zhejiang University, June 2000 to June 2002

Research Assistant, Department of Control Science and Engineering, Zhejiang University, September, 1997 to March, 2000

Name: Feng Xie Date of Degree: December, 2004

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: APPLICATION OF MODEL PREDICTIVE CONTROL TO A GEOTHERMALLY HEATED BRIDGE DECK

Pages in Study: 125 Candidate for the Degree of Master of Science

Major Field: Chemical Engineering

Scope and Method of Study: The purpose of this study was to design and implement a Model Predictive Control based control system for the Oklahoma State University Geothermal Smart Bridge Project. The control algorithm presented in this thesis utilizes weather forecasts generated by the National Weather Service Rapid Update Cycle forecasting model. Weather forecasts are used to determine when snow or ice accumulation is likely to form on the bridge surface. Weather forecasts are also used in conjunction with a first principle bridge deck model to predict the future bridge surface temperature trajectory. Weather measurement data for the bridge site was provided by the Oklahoma Mesonet. A cyclic with line search optimization algorithm implemented as part of this work calculates control moves by minimizing a quadratic objective function with linear constraints. Object Oriented Programming was extensively used in control system software development.

Findings and Conclusions: The work reported in this thesis represents a major advance in heated bridge technology. The control systems used in the previous heated bridge projects utilized On/Off feedback techniques with local weather information. The control system developed in this thesis utilizes advanced weather forecast and model predictive control techniques. The advantage of using this control system is that the heating system is engaged before the icing conditions reach the bridge and the bridge temperature is increased with optimized heat inputs. As a result, the bridge can be pre-heated to a desired temperature before the icing events, which guarantees the ice-free bridge condition. Real time results presented for four icing events during the winter of 2003-2004 demonstrated successful application of the control system. System performance proved most sensitive to the availability of weather forecast and measurement data. Extended delays of either compromised the ability of the control system to adequately preheat the bridge prior to an icing event. Improving the reliability of forecast and measurement data is the highest priority for future work. Improved optimization weighting factors were developed using data collected during one of the four icing events. Proper selection of the optimization weights over the prediction and control horizons proved critical to following the desired bridge temperature trajectory.

ADVISER'S APPROVAL: James R Whiteley