A DATA-DRIVEN APPROACH FOR SYSTEM APPROXIMATION AND SET POINT OPTIMIZATION, WITH A FOCUS IN HVAC SYSTEMS

by

Xiao Qin

A Dissertation Submitted to the Faculty of the

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

In Partial Fulfillment of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

2014

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Dissertation Committee, we certify that we have read the dissertation prepared by Xiao Qin, titled A Data-Driven Approach for System Approximation and Set Point Optimization, with a Focus in HVAC Systems and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy.

Date: 18 April 2014

Susan Lysecky

Date: 18 April 2014

Jonathan Sprinkle

Date: 18 April 2014

Ricardo Sanfelice

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copies of the dissertation to the Graduate College.
I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

Date: 18 April 2014

Dissertation Director: Susan Lysecky

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at the University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Xiao Qin

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisors Prof. Susan Lysecky and Prof. Jonathan Sprinkle for their guidance, encouragement, and support throughout my whole study. With them this work would not be possible.

I would also like to acknowledge National Science Foundation I-Corps program for both their financial support and the offering of an eye-opening workshop.

Lastly, I would like to thank my parents, family and friends. Your love and support carried me all the way.

TABLE OF CONTENTS

TABLE OF CONTENTS – *Continued*

TABLE OF CONTENTS – *Continued*

LIST OF FIGURES

LIST OF FIGURES – *Continued*

LIST OF TABLES

NOMENCLATURE

Symbol	**Description**
	HVAC system
x	Building indoor temperature
d	Building outdoor temperature
$\hat{x}$	Building indoor temperature estimation
Δx_d	Building indoor temperature change due to heat transfer (indoor-outdoor temperature difference)
Δx_w	Building indoor temperature change due to HVAC operation
σ	System disturbance
t^*	Averaged difference between indoor and outdoor temperature
r	HVAC set point temperature
r^*	Optimized HVAC set point temperature
$u(t)$	Binary control signal to HVAC system
w	HVAC utilization (in percentage)

Continued on next page

Nomenclature – continued from previous page

Symbol	Description
$\tilde{w}$	Estimated HVAC utilization (in percentage)
r^c	HVAC cooling set point temperature
w^c	HVAC cooling utilization (in percentage)
r^h	HVAC heating set point temperature
w^h	HVAC heating utilization (in percentage)
EQ	Abbrevation for Equilibrium sections
RU	Abbrevation for Ramp Up sections
RD	Abbrevation for Ramp Down sections
$\tilde{w}^{EQ}$	Estimated HVAC utilization in Equilibrium sections
$\tilde{w}^{RU}$	Estimated HVAC utilization in Ramp Up sections
$\tilde{w}^{RD}$	Estimated HVAC utilization in Ramp Down sections
$\tilde{w}^{RU_{T_0}}$	Estimated nominal HVAC utilization in Ramp Up sections
$\tilde{w}^{RD_{T_0}}$	Estimated nominal HVAC utilization in Ramp Down sections
T^{RU}	Transition time for set point to settle in Ramp Up section
T^{RD}	Transition time for set point to settle in Ramp Down section
Network Flow Model	
a_i	Set point temperature candidate
$t(i)$	Time layer i

Continued on next page

Nomenclature – concluded from previous page

Symbol	**Description**
$arc(i,j)$	arc connection between $node(i)$ and $node(j)$
x_{ij}	Decision variable on $arc(i,j)$
C_{ij}	Cost associated with $arc(i,j)$
D_{ij}	Comfort difference (penalty) associated with $arc(i,j)$
$b(i)$	Netflow balance on $Node(i)$
$transit(i,j)$	Transition indicator variable
$tran$	Total number of transitions
B	Budget constraint
TR	Transition constraint (limit on total number of transitions)
C_e	Cost information

ABSTRACT

Dynamically determining input signals to a complex system, to increase performance and/or reduce cost, is a difficult task unless users are provided with feedback on the consequences of different input decisions. For example, users self-determine the set point schedule (i.e. temperature thresholds) of their HVAC system, without an ability to predict cost—they select only comfort. Users are unable to optimize the set point schedule with respect to cost because the cost feedback is provided at billing-cycle intervals. To provide rapid feedback (such as expected monthly/daily cost), mechanisms for system monitoring, data-driven modeling, simulation, and optimization are needed. Techniques from the literature require in-depth knowledge in the domain, and/or significant investment in infrastructure or equipment to measure state variables, making these solutions difficult to implement or to scale down in cost.

This work introduces methods to approximate complex system behavior prediction and optimization, based on dynamic data obtained from inexpensive sensors. Unlike many existing approaches, we do not extract an exact model to capture every detail of the system; rather, we develop an approximated model with key predictive characteristics. Such a model makes estimation and prediction available to users who can then make informed decisions; alternatively, these estimates are made available as an input to an optimization tool to automatically provide pareto-optimized set points. Moreover, the approximation nature of this model makes the determination of the prediction and optimization parameters computationally inexpensive, adaptive to system or environment change, and suitable for embedded system implementation. Effectiveness of these methods is first demonstrated on an HVAC system methodology, and then extended to a variety of complex system applications.

CHAPTER 1

Introduction

1.1 Overview

Heating, Ventilation and Air Conditioning system, or HVAC system, is a system designed for obtaining a desired temperature and/or comfort level for a targeted space, usually indoor or vehicular, ranging from a simple stove in a home environment, to a complicated and reliable air-conditioning system in airplanes and submarines.(McDowall, 2006) As the name suggested, an HVAC system typically is capable of offering some (if not total) control of temperature, humidity, ventilation, and/or filtration, to change the target's air temperature and quality, with its operation based on the principles of thermodynamics, fluid mechanics, and heat transfer.

Among the various HVAC systems used in different places for variant purposes, the most common system that people use in their daily lives is a building HVAC system, which keeps the indoor atmosphere comfortable for their house and/or work place. Almost every building HVAC system up-to-date requires some input from human users that controls how the HVAC system should work. The typical input signal is a setpoint temperature that specifies the target temperature the building should obtain. No matter how "intelligent" modern HVAC would be, these decisions are, and will be, made by the users.

However, the end users are often found in a situation of making these decisions without having "enough" information to fully understand their consequences. For example, when setting the target temperature, users usually do not have the ability to gain insight in how much these settings would cost (in terms of electricity bill or

KWh usage), neither do they have the insight in how to set these target temperatures such that the cost of HVAC operation would be reduced while maintaining a certain level of comfort, unless “some information” on the prospective cost consequences is available.

Similar examples also exist outside the energy system world. For example, when a user is planning a route including a set of points in a city, and he/she wants to pick from a set of candidate routes that could potentially minimize the travel time. If we assume we know the speed of the user’s vehicle all the time and there’s no traffic at all, then this problem simply collapse into a linear programming problem. However, if the traffic is presented and is a spacial-temporal non-linear function, then the user is hard to make the optimal decision without knowing the traffic model.(Chiu et al., 2011; Hu, 2013)

These types of problems can be captured as a class of problems where users are asked to make decisions as inputs to a complex system (HVAC setpoint temperature, routes), to minimize a certain cost (electricity usage, traffic congestion) or to maximize a reward (comfort, arrival time), which may not be possible if some information on the system’s behavior (prospective cost, traffic) is not available. Therefore, mechanisms are needed to ensure effective feedback is available to users interacting with complex systems to aid on making optimal/sub-optimal decisions.

However, it’s not trivial for users to obtain these necessary feedback information on their own, because complex systems like HVAC operation are not trivial to model. It is true that modeling and simulation of these systems have been studied intensively both in academia and industry, however these models usually require expensive investments on the infrastructure and/or domain-specific expertise (Nassif et al., 2008; Peng and a.H.C. van Paassen, 1998; Riederer et al., 2002a,b; Tashtoush et al., 2005).

In this work we discuss data-driven methods to approximate such complex systems through linearization and machine learning, based on data sampled from *in-*

expensive sensors, and methods to calculate optimal/sub-optimal control signals for these systems based on the approximated system model. Unlike most of the existing approaches, this work is not to build a detailed model that potentially captures every aspects of the system, but an approximation model that captures the principle component of the system and provides estimations and predictions of the system behavior based on the input control signal.

A new concept thermostat/framework is developed in this work to demonstrate the proposed methods within the context of the HVAC domain and provide users with feedback information on their energy decisions, as well as suggestion on how they might be able to optimize these decisions, through a system approximation process as well as control signal optimization process, based on data collected from inexpensive sensors.

1.2 Motivation

A basic HVAC system consist of several core components including air damper, mixing chamber, heating/cooling coil, fan, etc.(ASHRAE, 1996; McDowall, 2006) Different configurations of these components would affect the performance of HVAC system dramatically, however, in this work we assume the HVAC system as a black box model, where the parameters of these internal components are determined and unchanged once the HVAC model is selected. Component configuration and optimization of the HVAC system is another topic in HVAC system design which is out of scope of this work.

The invention and development of HVAC system can be traced back to as early as 1880s when refrigeration became available. However, almost two centuries later there are still many areas of a modern HVAC system design that remain in active research and exploration, among which the *Energy Conservation* aspect of an HVAC system is a rising and most challenging one, with the goal to reduce the energy consumption of an HVAC system without compromising the comfort level.(Howell et al., 2009)

Numerous efforts are driving the research and development of technologies for HVAC energy conservation. For example, from the HVAC system operation perspective, (Lu et al., 2004, 2005) discussed HVAC system optimization in the condenser water loop, while (Fong et al., 2006) discussed HVAC system optimization from a evolutionary programming approach. Another example is the development of smart grid technologies (Electricity Advisory Committee (EAC), December 2008; U.S. Department of Energy, 2008). In this broad scope there are technologies ranging from low level innovations to efficiently harvest power from renewable resources (Darghouth et al., 2011; Keis et al., 2002; Wang et al., 2007) to higher level mechanisms that efficiently monitor and manage the underlying computational and physical resources (Darghouth et al., 2011; Jiang et al., 2007; Lien et al., 2009; Sundramoorthy et al., 2011). Ground level programs are also appearing, striving to engage end users to consider how they too can contribute to becoming more energy efficient. For example, Autodesk Green Building Studio (Autodesk Inc., 2010) is a web service that analyzes a building model and provides a baseline report on the proposed building's carbon output from the consumption of resources such as fuel, electricity, or water. Simulation tools are also available such as the EnergyPlus simulation program (Department of Energy, 2012), to calculate hourly energy cost for a variety of commercial and residential buildings. Parameters such as the building's construction and climate are considered, and integrate low-level HVAC and duct loss models, among others.

HVAC energy conservation technologies are also of interest to end users. From a homeowner's standpoint, reduction of electricity consumption is beneficial in direct monetary savings. According to the EPA report (US Environmental Protection Agency (EPA), 2012), the average US household spends more than $2,200 a year on energy bills, with about half of this amount incurred from heating and cooling. From a national energy consumption perspective, home energy consumes 25% of the total energy consumed in the U.S. and worldwide.

HVAC systems are usually (if not always) equipped with a thermostat, a device acts as the user interface that takes input from users and generate output to control the HVAC's components to obtain the desired temperature and air quality. It can be as simplest as setting the temperature to a fixed target through a “knob”, to a complex embedded system or SoC based“smart thermostat” with various features.

Figure 1.1 shows a smart thermostat that is equipped with a feature that enables users to program set point temperatures for different time slots, such as “Morning”, “Work”, or “Night.” through a LCD touch screen as well as web-based user interface. (Filtrete, 2013)

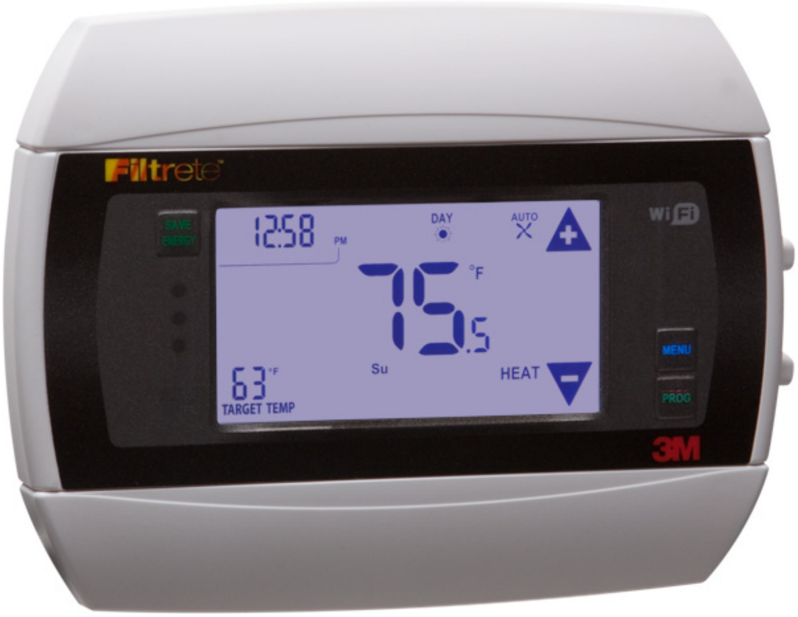


Figure 1.1: Smart Thermostat with Programmable Features

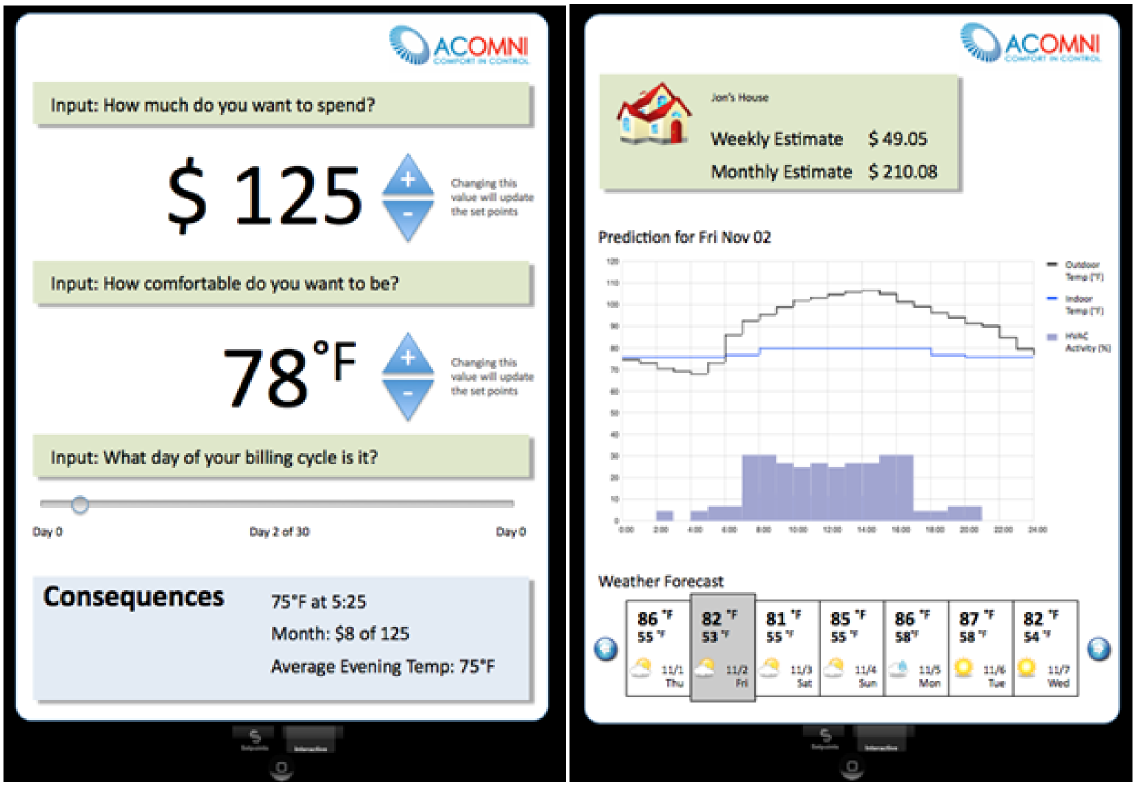
While these thermostats provide a homeowner with time-indexed control over the HVAC operation, homeowners are ultimately responsible for synthesizing the setpoint array to meet their personal preferences for cost savings and comfort level. Gao et al (Gao and Whitehouse, 2009) found that many users do not take advantage of the programmable features, because users have difficulty determining a schedule on their own. To help homeowners to more efficiently utilize energy, utilities are increasingly offering energy management services (Tucson Electric Power (TEP), 2012b) by providing remote monitoring and control hardware, as well as shared control of HVAC operation with the utility. New thermostats are also available which integrate environmental sensors (e.g. human activity and temperature) and employ machine learning mechanisms to determine the occupant's habits, and switch HVAC modes based on these routines (NEST Lab, 2011).

However, it is not trivial for a homeowner to predict how much of an impact a decision, such as adjusting the thermostat set point, may have. To further complicate matters, homeowners may need to consider savings that can be obtained by reacting to dynamic pricing policies based on the time-of-usage such as peak time (most expensive), shoulder time (moderate) and/or off-peak time (inexpensive) (Tucson Electric Power (TEP), 2012a), biasing scheduling of high-energy systems to run at non-peak times. From an end user's perspective energy consumption is invisible and abstract, leading to a poor understanding of how much energy appliances consume, as well as misconceptions in how to conserve energy (Froehlich, 2009). Moreover, the operational status of the HVAC system (either "On" or "Off") is emergent, based on the homeowner's selected set point temperature and environmental factors. Thus, while numerous platforms are available that monitor energy usage, information in itself does not effectively change end user behavior, nor does it help the end users save energy/money. (Hargreaves et al., 2010).

This work introduce a new concept thermostat with a prediction feature which enables users to see the energy consequences of their decisions regarding comfort. More specifically, when a user sets or resets the set point temperature schedules of a building (home), the thermostat would provide a instant feedback on how much energy this setting (new setting) will cost within a predictive time horizon (*e.g.*, $N = 30$ days), thus gives the user a "tangible" feedback of their energy decisions. In addition, the thermostat would be equipped with an additional optimization engine that optimizes the set point temperatures based on a user cost input, acting as a recommender system to provide user with alternative method to determine the setpoint schedule.

In this scenario, we remove the need for a human-entered array of time-varying set points. Instead, the goal is to have a single input, which specifies the desired HVAC operation costs for a time horizon (e.g., $N = 30$ days). For any non-negative cost, though, the homeowner may wish to understand the internal home temperatures that are the predicted consequence of this decision. As an example, a homeowner may be willing to spend \$200/N to keep internal temperature below 82°F, but may not be willing to spend \$250/N to keep the internal temperature below 80°F.

With this proposed new interfaces and tools that enable end users to visualize the long term energy and cost consequences of their decisions and provide aid to the end users for their temperature settings, the end users would ultimately be able to consider themselves sufficiently informed to make energy decisions. Moreover, the interactive cost feed back information would help save energy (Fischer, 2008), and invoke a behavior modification impact to the users by bringing energy-awareness incentives to the users to save energy. An example user interface implemented on an mobil platform is shown in Figure 1.2, where Figure 1.2a illustrates a user interface for real-time cost-comfort tradeoff which enables user to interact (*i.e.* change setpoint temperature or cost constraint) and observe the tradeoff, while Figure 1.2b illustrates an interface for cost estimation with a proposed setpoint temperature schedule in a prediction horizon.



ACOMNI COMFORT IN CONTROL

Input: How much do you want to spend?

$ 125

Changing this value will update the set points

Input: How comfortable do you want to be?

78°F

Changing this value will update the set points

Input: What day of your billing cycle is it?

Day 0 | Day 2 of 30 | Day 0

Consequences

75°F at 5:25

Month: $8 of 125

Average Evening Temp: 75°F

(a)

ACOMNI COMFORT IN CONTROL

Jon's House

Weekly Estimate $ 49.05

Monthly Estimate $ 210.08

Prediction for Fri Nov 02

Weather Forecast

11/1 Thu	11/2 Fri	11/3 Sat	11/4 Sun	11/5 Mon	11/6 Tue	11/7 Wed
86 °F	82 °F	81 °F	85 °F	86 °F	87 °F	82 °F
55 °F	53 °F	55 °F	55 °F	58°F	58 °F	54 °F

(b)

Figure 1.2: (a) Example user interface of cost feedback on setpoint temperatures (b) Example user interface of cost prediction

1.3 Potential Impact

1.3.1 Behavioral Modification Effect

A typical user of the proposed innovation could be a homeowner that interacts with a thermostat to set up the setpoint temperatures for the home HVAC system. It could, however, expand to other archetypes of end users who bears the same behavior such as building managers and apartment managers.

Currently most users choose their target setpoint temperatures in a “open loop” fashion in terms of cost, they are either unconscious of the corresponding cost or lack of proper tools to gather the cost information, leaving them in the dark trying to optimize the setpoint, *e.g.* a common approach is to utilize a cost-saving setpoint sequence, where modes such as "Work" or "Evening" provide an ability to change the setpoint for comfort when the homeowner is home, and reduced energy usage when not. However, these empirical "hard-coded" optimization attempts are usually non-optimal due to their open loop nature, and the users have no incentives to reduce the cost due to lack of necessary feedback on the energy usage or cost.

With the proposed work, a candidate series of setpoints (created by the homeowner) can be applied to estimate usage and cost. Alternatively, cost can be the input, and the series of setpoints can be shown. Then, the customer can decide whether the anticipated discomfort of the setpoints is worth the monetary savings. If the homeowner ever decides to change the setpoints or goals, the cumulative costs are updated in the interface.

The proposed innovation is able to help users making a well-informed decision, or even reform the way the users interacts with systems like HVAC. Equipped with real-time feedback information of their decision consequences, the users may eventually change the way they used to set their target temperatures, which conceptually, can be considered as a behavioral modification effect. This innovation brings energy-awareness to the users, and bring them incentives to reduce the home energy consumption.

1.3.2 Scalability

In addition to being a framework that puts the cost of HVAC operation directly into the hands of the homeowner, the proposed innovation has the ability to be scaled up and applied at a higher level energy eco-systems, requiring little modification. Potential topics include demand side management, neighborhood/community optimization, and renewable energy integrated optimization.

Demand side management, or energy demand management, is a process of modifying consumer-side energy demand to address the peak power problem. Typically incentives or education are used to encourage users to lower energy usage during peak hours, or move energy usage to off-peak times. A more evenly distributed load profile would have a higher efficiency on the energy generation site. From the utility's perspective, feedback information of the load profile could aid in making decisions such as shutting backup generators to save energy or starting them to fulfill a predicted high volume energy usage. Additionally, a dynamic pricing approach could be utilized (Department of Energy, 2005) to encourage users change their energy usage. Users similarly benefit from real-time feedback information on their energy systems, as the users could make a well-information decision or have an optimizer to suggest an optimal solution (*i.e.* HVAC target temperatures, appliances operation time etc.), which could potentially reduce the cost of running these systems.

This framework can be scaled to include multiple households into the same picture as well. Besides running system feedback and optimization for a single household, a community level framework could gather feedback from each individual house and run optimizations on the community level as a whole.

Moreover, with the development of renewable energy resources and their appearances in home building (such as solar panels), new challenges arises on energy optimization with these energy generation units. The proposed work could potentially works for these systems as well with minor modifications on the system modeling approaches and integrating feedback information as well as optimization with renewable energy resources.

1.4 Contribution

In this dissertation, a data-driven approach for HVAC system modeling, prediction, and optimization is proposed. It has the following features that differentiate itself with existing approaches:

- It employs a data-driven approach, with *inexpensive* sensor gathering necessary data, and machine learning algorithms to capture the principle operation model of the system, while most of the existing approaches require significant investment in sensor infrastructure and/or domain expertise to model a complex system. Because of its data-driven nature, the proposed work does not require users to manually input parameters that might be specific to a target building, instead, all the parameters will be learned by the system through the machine learning process, which makes this approach generative to any buildings.

- The proposed work is able to provide a *mid-term* prediction/estimation of the HVAC system operational cost, which typically has a 10-day prediction scope, while other existing methods usually have a much shorter prediction scope ranging from minutes to hours.

- The accuracy on the system cost prediction of the proposed approach is above 90%, moreover, it maintains a high accuracy of prediction regardless of the HVAC load factor. *i.e.* Regardless of whether the HVAC system is running heavily (for example cooling in summer) or idling most of the time (cooling in winter), the system could effectively predict the cost of HVAC operations.

- An integrated optimization engine is also developed with the proposed work to calculate an optimal/sub-optimal setpoints to help the users make decisions. Furthermore, this optimization as well as prediction process can be performed "on-line", therefore give users an "up-to-date" information on the system operation cost as well as decision options.

- The proposed approach utilizes approximation and regression techniques to linearize a complex system, which makes it computational inexpensive and suitable for implementation on embedded devices such as a smart thermostat, thus ideal for integrating into existing infrastructure as well as emerging smart grid technologies.

- Finally, we illustrate how the proposed approach can be expanded to address a set of problems with similar properties, as well as energy systems with larger scope (community, regional), components (renewable resources) and constraints (dynamic pricing). It could shed some light on designing systems that brings immediate feedback information to users thus help them to make a well-informed decision.

1.5 Dissertation Organization

The dissertation is organized as following: in the remaining sections of Chapter 1 we discuss the motivation of work on HVAC systems, as well as introductions on the methodologies utilized in this work. This chapter finishes by discussing the impact and contribution of this work. Chapter 2 starts by introducing necessary background as well as literature reviews on existing approaches, and finishes by formulating the problem this work is trying to solve. In Chapter 3 we discuss the abstraction and generalization of this method, and provide guidelines to apply this method to other similar problems. Chapter 4 discusses the details of the proposed approach and its implementation for HVAC system modeling and prediction, where Chapter 5 discusses the optimization part of HVAC system problem. Their experimentation setup as well as result are discussed in Chapter 6. Finally Chapter 7 concludes this work an discuss the possible future works.

CHAPTER 2

Background

2.1 HVAC Modeling

Numerous efforts exist both in academia and industry to model an HVAC system, and can be generally divided into two categories: analytic approach modeling and data-driven approach modeling.

2.1.1 Analytic Modeling

Analytic modeling approaches for HVAC system modeling are usually based on principles of physics and thermodynamics, and applied to individual components and/or the overall HVAC system. It requires a valid mathematical description of a system model, which is derived from theoretical equations, and a set of parameters in these equations governing the characteristics of the model. These parameters are usually associated with the deployed HVAC system based on their configuration and datasheet, and are unique to a specific target building. Based on the building's physical attributes such as footprint, insulation material, location, orientation etc., individual parameters must be configured. Because the non-linear nature of the system as a whole, almost every analytic approach introduces some approximation to simplify the problem such as to reduce the complexity of the system modeling.

First principles modeling

First principles modeling derives the theoretical equations for the system and builds the model upon it. The building's model is developed from physics laws, and the knowledge of the building envelope and materials, as well as the building's thermal parameters are utilized. In (Yik et al., 2001) models are developed to predict the energy consumption for a wide range of commercial buildings, based on building characteristics such as floor area, air conditioning system type, and the year when the building was constructed.

A model to predict energy consumption based on the overall thermal transfer value (OTTV) is given in (Chirarattananon and Taveekun, 2004), and is derived from the building's wall composition, glazing types, wall-window ratio, among others. However, depending on its occupancy the thermal model of a building can vary significantly (ASHRAE, 2001). As in (Gugliermetti et al., 2004), the influence of the climate model is also needed to predict building energy consumption.

(He et al., 1997) presented a lumped-parameter model for the dynamics of the vapor compression cycles. (Jin et al., 2007; Wang et al., 2004) presented a cooling coil unit model based on energy balance and heat transfer principles, and reported a simple but accurate prediction model for HVAC control. State space is also used in (Kulkarni and Hong, 2004) to model the building dynamics and control the HVAC.

Simulation based modeling

Simulation based modeling approaches can also be found in literature where HVAC system simulation software packages were utilized for modeling. Technically, these software packages are using an analytic modeling approach as well to providing users a system model, where the users will be prompted to input the parameters and configurations of the to-be-modeled system. Users can also provide both spacial and temporal simulation of the target system.

The Alternative Energy Product Suite (AEPS) System Planning tool (Alternative Software Concept, 2012) is a software application that focuses on the design, modeling, and simulation of electrical energy systems with an emphasis on renewable energy sources (solar, wind, and hydro). These tools calculate energy generation, consumption, and storage for modeled systems. Energy and cost data can be analyzed to optimize the modeled system based on user objectives and priorities. REM/Design^{TM} similarly calculates heating, cooling, domestic hot water, lighting and appliance loads, consumption, and costs based on a description of the home's design and construction features as well as local climate and energy cost data (Architectural Energy Corp., 2012).

TRNSYS (University of Wisconsin Madison, 2012) is an energy simulation program taking a modular system approach that utilizes a system description language to enable a user to specify platform components and the manner in which they are connected. Due to its modular approach, TRNSYS is extremely flexible for modeling a variety of energy systems in differing levels of complexity. However, no assumptions about the building or system are made (although default information is provided) and it is up to the end user to provide detailed information about the building and sub-systems.

Additional literature utilizing simulation based modeling can be found in (Clarke et al., 2002; Cui et al., 2008; McDowell et al., 2003; Sowell and Haves, 2001; Zhou et al., 2008)

Limitations

Although the analytic approaches can offer a valid modeling tool of the target system, they have the following drawbacks and limitations:

- A theoretical model for the target system as well as the model parameters is needed. Often the process of choosing the suitable model and obtaining its parameters requires some expertise in the HVAC system domain, and is not typically a feasible task for a homeowner.
- Some analytic models only exist at the component level. *i.e.*, they are not suitable for a system level modeling.
- The model is vulnerable to dynamic changes in the parameters and variables,

which could bring inaccuracy or even incorrectness to the results of the system model output.

- Developing a model is usually computational expensive.

To overcome these difficulties, data-driven approaches are applied.

2.1.2 Data-driven Modeling

Data-driven approaches have been well documented in literature as well as industries in the past decades with the development of new data mining and machine learning algorithms. Methods deploying data-driven approach will usually requires a set of data that is either sampled on-line or off-line as the training data for model building. The most common data-driving approaches utilized in modeling HVAC systems are parameter estimation modeling, time-series modeling, regression modeling, and Artificial Neural Network modeling.

Parameter Estimation Models

For a given approximation of a building model, parameters can be estimated and fit based on historical data. In (Wang and Xu, 2006) genetic algorithms are used to fit models based on lumped parameters. The model was validated by comparing predicted and actual indoor air temperature over time.

(Sturzenegger et al., 2012) propose automated linearization and parameter estimation of thermal dynamics in order to permit composition of individual rooms to produce a building model. Those results permit HVAC actuators, and alternatives such as window blinds, to be considered when predicting heat flux.

Regression Models

In (Catalina et al., 2013) the heating energy demand of a building was predicted by a multiple regression model, where there inputs for this model are building global heat loss coefficient (G), the south equivalent surface (SES) and the difference between the indoor set point temperature and the sol-air temperature, and this model was validated with data sampled from 17 blocks of flats. (Katipamula et al., 1998) employed a multiple linear regression (MLR) models to derive baseline models and detect deviations in energy consumption under variant operational conditions. In (Wu and Sun, 2012a,b) a physics-based auto-regression moving average (pbARMAX) model is used for indoor temperature prediction. (Leung et al., 2012) uses weather temperature data as well as the occupancy space electrical power demand as inputs to a neural network to predict the building cooling load, where the prediction yields acceptable results for summer seasons, large variation is presented for winter seasons. (Yun et al., 2012) use an indexed ARX model, where the coefficients of the model change according to time, to produce a 1-hour ahead prediction of building thermal load. (Escrivá-Escrivá et al., 2011) uses ANNs on independent end users to obtain a prediction of electricity consumption by summing each end users' consumption. (Mustafaraj et al., 2010) used Box–Jenkins (BJ), auto-regressive with external inputs (ARX), auto-regressive moving average with external inputs (ARMAX) and output error (OE) models to produce a 30-min or 2-hour ahead prediction of room temperature and humidity, and validate the models by sampled data in weekdays of summer, fall and winter season. (Li et al., 2009) investigated four modeling methods, namely back propagation neural network (BPNN), radial basis function neural network (RBFNN), general regression neural network (GRNN) and support vector machine (SVM) for hourly cooling load prediction. (Xi et al., 2007) employed support vector regression (SVR) to build a nonlinear dynamic model of a HVAC system, upon which a non-linear Model Predictive Controller was design. (Dong et al., 2005) employed support vector machines with radial-basis function (RBF) kernel for building energy consumption prediction. With inputs from outdoor temperature, relative humidity as well as global solar radiation, they reported a prediction result with less than 4% error.

Artificial Neural Network models

An artificial neural network (ANN) is used to determine the corresponding building energy consumption model (Beghi et al., 2010; Ben-Nakhi and Mahmoud, 2004; Karatasou et al., 2006a). In (Karatasou et al., 2006b), time series analysis was combined with ANNs to predict 1-hour-ahead energy consumption. In (Yang et al., 2005), two adaptive ANN models were proposed to predict the energy use, namely the accumulative training and sliding window training models. An ANN is trained in (Gonzalez and Zamarreno, 2005) by means of a hybrid algorithm where prediction is made based on current and forecasted values of temperature, current load and the hour of the day. Alternatively, (Amjady, 2001) utilizes a prediction method with auto-regressive integrated moving average (ARIMA) with results demonstrating a better fit than an ANN. In (Cherkassky et al., 2011), a clustering and regression method is used to predict the (electrical) power consumption, dividing the data into occupied, unoccupied, ramp-up and ramp-down sections where regression is used on the occupied and unoccupied segments to predict the power consumption. (Li and Huang, 2013) reviewed and evaluated four popular models for short-term cooling load prediction, including Auto-regressive Moving Average with Exogenous inputs (ARMAX) model, Multiple Linear Regression (MLR) model, Artificial Neural Network (ANN) model and Resistor–Capacitor (RC) network. With TRNSYS simulated data the MLR model and the ARMAX model have better prediction accuracy and precision and the RC network model has better adaptability to control set points. (González and Zamarreño, 2005) deployed a feedback ANN to predict the hourly energy consumption in buildings. (Peng and a.H.C. van Paassen, 1998) used a zone model derived from computational fluid dynamics to model the indoor temperature as a state space model. (Chen et al., 2010; Guan et al., 2009) utilized wavelets transform as well as neural network to make short term load prediction, where historical and similar day load profile were feed to wavelets transforms for frequency decomposition, then individual neural networks were deployed to predict each frequency component, which was summed to get the final prediction.

Limitations

While a data-driven approach generally alleviates the building-specific dependency, it also has limitations:

- In order to achieve a desired accuracy these approaches may be computationally expensive, making them unsuitable for an embedded system implementation (Stanek et al., 2001).
- Many of these approaches focus on the individual error at each sample point, rather than taking the error distribution as a whole into consideration, therefore lacks a holistic view on the model over time.
- Many of these approaches require an intensive data sampling process, which leads extra cost on sensors and infrastructure if not already installed.
- The prediction horizon for these approaches are relatively short (in some, as short as 1 hour); such a time horizon is not sufficient for a 7-10 day cost accumulation, or to give users a meaningful feedback.

2.2 Usage optimization

HVAC system optimization often refers to improve the system efficiency and/or reduce the cost of running the system, and numerous efforts have been found to obtain this goal. Depending on where the optimization occurs they can be categorized as component level optimization and system level optimization. Component level optimization aims at improve the HVAC system efficiency by improving one or more components in the system (Jin et al., 2007; Ke and Mumma, 1997; Wang et al., 2004). On the other hand, system level optimization often involves changing the control schema of the HVAC system while leave each of the components intact

(Mossolly et al., 2009; Nassif et al., 2004, 2005). In this dissertation we mainly focus on the second type of optimization and study the effects of manipulating set point temperatures to the HVAC system efficiency. This work treats the HVAC system as a black box, where the input to it is the set point temperature, the output is the operation status of the system, and the cost is a function of its operation status and time. under these circumstances optimization means finding an optimal control (set point temperature) strategy that reduce the cost of running HVAC system while maintaining a certain level of comfort for the users.

2.2.1 Open-loop control

The most straight-forward approach to control the HVAC system input (set point temperature) comes with an “Open-loop” or “fire and forget” fashion, where a user configures a set point temperature and lets the system run under this configuration. This control scheme is what most of the current thermostat and HVAC system adopts, including the simple “knob” form thermostat where the user configures a single set point temperature for hold all the time until it’s changed manually by the user again, and the “programmable” thermostat where user can specify the set point temperatures for different time of the day such as “work”, “home” and “sleep” etc. No matter what forms of input are provided to the thermostat, these control schemes do not provide users any feedback on their set points cost-wise, and therefore considered as an open-loop control scheme. The HVAC/thermostat itself is a close-loop control in terms of driving the temperature to the set point.

Studies have shown that users are not good at choosing the optimal set point temperatures (Gao and Whitehouse, 2009) to reduce the cost, and the “programmable” thermostat are often found misleading as these programmable features make the users think they are saving energy. Indeed, only with carefully selected set point temperature can a smart thermostat help save energy. Some “smart” thermostats take a more active role to detect the presence of human in the building with motion sensors and turn on/off the HVAC system accordingly, while other thermostats have "learning" abilities to acquire the user's set point temperature and presence pattern statistically. There also exists smart thermostats that will perform pre-cooling and pre-heating actions to optimize the energy usages, i.e. the thermostats will do horizon prediction that a set point change is coming, and start cooling/heating in advance, or reduce energy consumption by doing advanced scheduling of the HVAC unit to take advantage of the known best performance of those units. These further steps may help reduce the energy usage to a certain degree depending on the accuracy of these sensors and the occurrences when the user is absent. However, none of these smart thermostats provide a cost-comfort correlation to the users to show them the trade-off, nor the consequences of their energy decisions, not to mention an optimization of set point schedule under a cost/consumption constraint.

2.2.2 Application to specific building models

Given a building-specific thermal model (Fong et al., 2009; Mathews et al., 2000; Nassif et al., 2004) a number of tools can then employ optimization algorithms such as reset control, genetic algorithms, etc. to optimize the HVAC operation. Determining these values are cumbersome even for a technically savvy homeowner, so alternative approaches to use collected data have been proposed. These approaches typically collect the thermal data from a building and employ an optimal search algorithm in the configuration space, to determine the optimal solution given a specific thermal configuration (set point, outside temperature etc) (Wemhoff, 2010).

There are a few limitations when applying this approach to choosing new set points. First, the collected data may not cover the complete configuration space, so algorithms designed to find an optimal solution are unlikely to choose values outside the region previously explored. Moreover, many of the previous efforts that consider HVAC optimization employ short-term HVAC usage prediction, limiting the scope of the optimization to one or two days.

2.2.3 Predictive control

Model Predictive Control (MPC) (Camacho and Bordons, 2004) and Nonlinear MPC (Allgöwer and Zhen, 2000) have been successfully applied to many long horizon control optimization problems due to its ability to meet constraints without requiring expert intervention (Garcia et al., 1989). In climate control optimization applications, the MPC approach is to guarantee certain comfort levels at predetermined set points. The use of MPC for optimization of a given cooling demand is described for buildings with a series of chillers is discussed in (Ma et al., 2012). That approach uses a hybrid model for cold water storage tank, is validated by historical data, and takes into account weather prediction to predict future states. In (Oldewurtel et al., 2012) the approach also utilizes weather prediction, and demonstrates accuracy and tunability of an MPC approach across many different kinds of buildings and HVAC systems. Authors in (Aswani et al., 2012a,b) introduce a way of using a semi-parametric regression to estimate the building heating load, and apply learning-based MPC to reduce the energy consumption in a single room testbed. In multi-zone office buildings, the feasibility of an MPC approach is discussed in (Privara et al., 2011). MPC has also been shown to be successful in emphasizing energy savings with minimal retrofit needs, as in (Široký et al., 2011). In (Avci et al., 2013) a Model Predictive Control (MPC) method is utilized to optimize the HVAC system operation cost as well as the user comfort level with real-time electricity pricing policy. The indoor temperature was modeled using a first-order linear system ($\dot{x} = f(x)$) and the setpoint temperature was selected based on electricity price as well as a user tolerance factor. MPC was utilized based on the indoor temperature model and a simulation on a 48-hour data was carried out to validate the controller decisions.

The benefits of MPC are evident in the literature, but still require a definition of the state update function $\dot{x} = f(x, u)$ (either in continuous or discrete form) in order to optimize the system cost over time. Such an approach requires installation of sensors and other equipment whose cost and energy to manufacture may exceed the savings gained by the optimization approach. Without specification of the state update equation it is not possible to directly apply an MPC approach, since the optimization phase usually requires a closed form expression for the state update equation, in order to utilize gradient descent (Richter et al., 2009) or other optimization methods.

2.3 Problem statement

Therefore in order to enable the user making "well-informed" decisions, feedback information on the correlation of their decision and cost should be presented. Moreover, optimization should be conducted based on this correlation to close the loop. This work presents the formulation of the correlation and optimization problem, as well as the methodology of solving such a problem. This methodology is different from other smart thermostat optimization problems in the sense of providing a predictive cost-comfort correlation as well as cost-constrained optimization. Moreover, in this paper we propose to avoid specifying the state update function due to the accumulated error over a long (*e.g.* 10 days) horizon of such a model.

Concretely, our approach requires (partial) sampling of state variables and inputs, and is decomposed into two tasks: prediction and optimization. The prediction task provides an estimate of cost for a future horizon, based on a recent input horizon. The optimization task uses the same input horizon, plus the cost estimate, to select set points in order to meet the cumulative cost constraints.

Figure 2.1 presents this framework applied to a user-centric HVAC management framework composed of prediction and optimization components which (1) dynamically capture the characteristics of the underlying system and estimates its behavior within a given time horizon, (2) calculates cost of operation based on the set point schedule, electricity cost (c_e), weather, and predicted HVAC usage, and (3) generates an optimized time-varying set point schedule r^* for the HVAC unit based on user defined constraints. To briefly describe the variables in the framework: $mode \in H, C$ means an HVAC can be working in one of two modes: Heating or Cooling, r is the user-set temperature while r^* is the optimized temperature set points generated by the framework; u is the control signal from thermostat to the HVAC unit; x is the indoor temperature while d is the outdoor temperature; Δx_w stands for the temperature changed due to HVAC work. $\tilde{w}$ is the predicted HVAC utilization, and C_e the cost constraint.

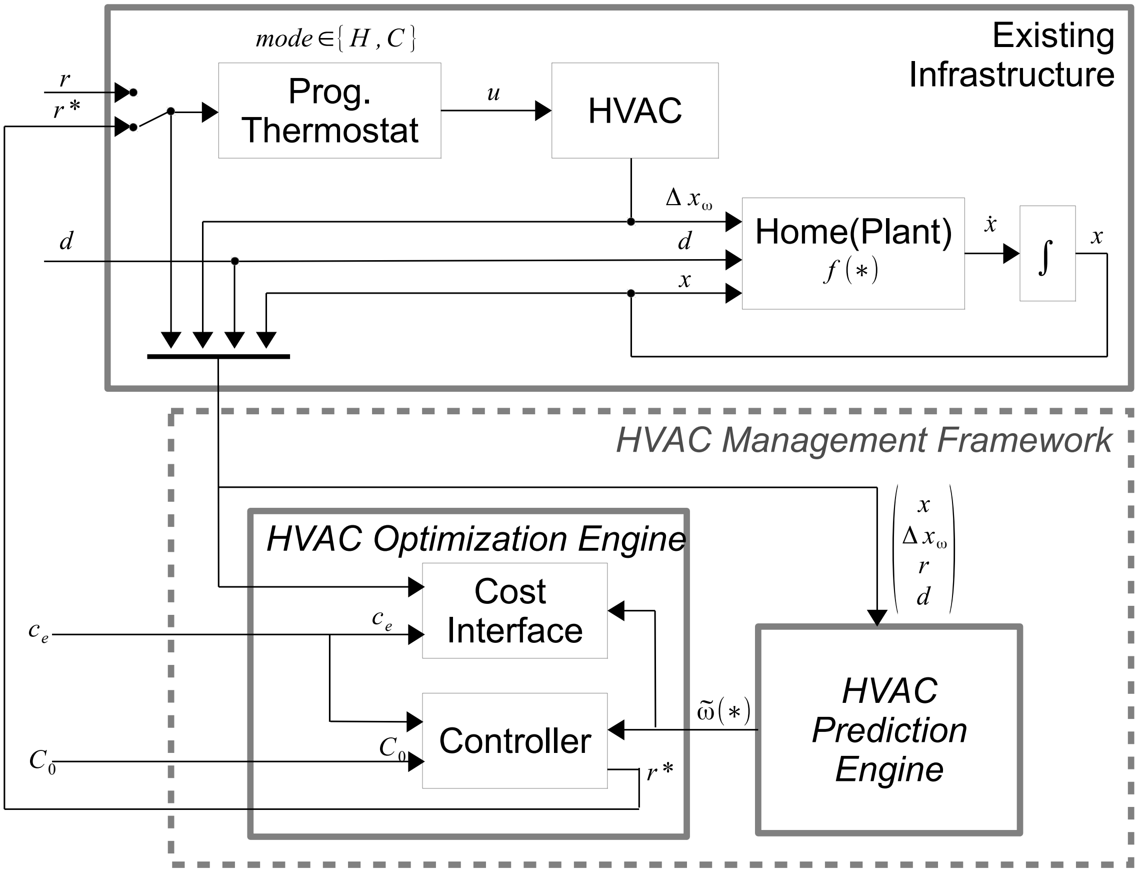


Figure 2.1: Block-level diagram of the Cumulative Cost Optimization Framework

We use the following variables: disturbance d represents the outside temperature, and x represents indoor temperature, r is the (time-varying) HVAC set point signal. Let $u(t) = u \in \{0, 1\}$ represent the off/on control signal to the plant, and let the cost of operating the plant over a continuous horizon from time 0 until time T be

$C = \int_0^T K u dt$ where $K \in \mathbb{R}^+$ represents the unit cost of plant operation.

The signal u is the result of a potentially nonlinear control function $u = g(x, r)$, where $x, r \in \mathbb{R}$ are also time-varying signals that represent the state value and set point signals, respectively. We assume a nonlinear function $\dot{x} = f(x, u, d)$ where d is a bounded, time-varying, Gaussian disturbance.

The correlation problem is to find a model M that maps the set point temperature r to the cost C, over a prediction horizon $[T_k, T_{k+n}]$, given data sampled from $[T_i, T_j]$, $i \in [0, k], j \in [i, k]$, and a prediction of disturbance over the prediction horizon d_i, $i \in [k, k+n]$. The problem is to generate r such that the actual cost, C, is within some neighborhood of a desired cost, C_0. The contribution of this work is to generate r *without explicitly* defining f in closed form. Rather, the work directly measures the integrated values of u over fixed horizons with known *measured* inputs of d and x, and with predicted values of d, but *without* predicting x over the horizon. This is possible *if* x tracks r for a large portion of the horizon: exceptions to this represent either guaranteed cost, or guaranteed cost savings. The horizon in this work is on the order of 7-10 days.

In this work we assume the system is in cooling mode, though equivalent statements for heating mode are straightforward. Due to space constraints, this paper discusses behavior during seasons where the temperature is higher than the set points during the day, thus we satisfy the requirement that x tracks r for a large portion of the horizon. This assumption does not reduce the impact of the work, since those seasons of the year often reflect the highest strain on homeowner cooling costs, and frequently strains utilities that must meet consumer demand, which may cause a brownout/blackout (FRONTLINE, 2010; Norman L. Miller and Auffhammer, 2008).

CHAPTER 3

Framework Abstraction

In the previous chapters a HVAC system modeling and optimization problem is introduced. In this chapter we extend the discussion to solving similar optimal control problems. We start with a brief review on the classic linear system optimal control theory, as well as a description of model predictive control, followed by discussions on their requirements of obtaining a system model. Then we provide an abstraction of a data-driven approximation approach and its general work flow, which relaxes the requirements on the system model, alleviating the computational demand, while maintaining an acceptable outcome.

3.1 Classic system control

3.1.1 State space model

In control theory, a state space representation can be used to model a linear system with state differential equation and output equation (Dorf, 1995). A typical state space model is written as:

$$\begin{cases} \dot{\mathbf{x}}(t) & = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \\ \mathbf{y}(t) & = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \end{cases} \tag{3.1}$$

where $\mathbf{x}(\cdot)$ is the system state vector, $\mathbf{y}(\cdot)$ the system output vector, $\mathbf{u}(\cdot)$ the control vector, and $\mathbf{A}(\cdot)$,$\mathbf{B}(\cdot)$,$\mathbf{C}(\cdot)$ are state, control, output matrix respectively. A non-linear system can be linearized to this form on some equilibrium or reference point.

For the purpose of simplicity we utilize a Linear Time Invariant (LTI) system without losing generality, also we assume the system output is independent with control signal, *i.e.* $\mathbf{D} = \mathbf{0}$, therefore the system becomes:

$$\begin{cases} \dot{\mathbf{x}} & = \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} & = \mathbf{Cx} \end{cases} \tag{3.2}$$

There are many advantages of representing a linear system with state space model. We point out some of these advantages without proof below, interested readers can refer to (Dorf, 1995) for details.

- System transfer function

  Laplace transform of system transfer function $G(s)$ can be calculated by $G(s) = \mathbf{C\Phi(s)B}$, where $\mathbf{\Phi(s)} = [s\mathbf{I} - \mathbf{A}]^{-1}$ is the Laplace transform of state transition matrix $\mathbf{\Phi(t)} = exp(\mathbf{A}t)$.

- Controllability

  State controllability implies that it is possible to steer the states of the system from any initial value to any final value within a finite time window. A system is controllable if the following equation holds:

$$rank([\mathbf{B}, \mathbf{AB}, \mathbf{A^2B}, ..., \mathbf{A^{n-1}B}]) = n \tag{3.3}$$

- Observability

  State observability is a measure for how well a system states can be inferred by the knowledge of external output. A system is observable if the following equation holds:

$$rank(\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA^2} \\ ... \\ \mathbf{CA^{n-1}} \end{bmatrix}) = n \tag{3.4}$$

Moreover, a Linear Quadratic Regulator (LQR) can be applied for system optimal control, as described in the following section.

3.1.2 Optimal control

Optimal control system is a system that provides a minimum performance index, such as the sum/integral of squared error. Concretely, the performance of a system can be written in terms of state variables $\mathbf{x}$ and control signals $\mathbf{u}$ as:

$$\mathbf{J} = \int_0^t g(\mathbf{x}, \mathbf{u}, t)\mathrm{d}t \tag{3.5}$$

Suppose the system is represented as

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} \tag{3.6}$$

and the performance index is defined as

$$\mathbf{J} = \int_0^t (\mathbf{x}^T\mathbf{x})\mathrm{d}t \tag{3.7}$$

The above performance index indicates the goal of the control system is to bring the internal states to the origin, *i.e.* $\mathbf{x} = \mathbf{0}$.

If a feedback controller where $\mathbf{u}$ is some function of the measured state variables $\mathbf{x}$ is used:

$$\mathbf{u} = -\mathbf{k}(\mathbf{x}) \tag{3.8}$$

we could define a matrix $\mathbf{H}$:

$$\mathbf{H} = \mathbf{A} - \mathbf{BK} \tag{3.9}$$

And the optimal control parameter can be obtained by:

1. Determine a matrix $\mathbf{P}$ that satisfies Equation (3.10), where $\mathbf{H}$ is defined in Equation (3.9)

2. Minimize $\mathbf{J}$ by determining the minimum of Equation (3.11) by adjusting the unspecified system parameters (*i.e.* $\mathbf{K}$ for this example)

$$\mathbf{H}^T\mathbf{P} + \mathbf{P}\mathbf{H} = -\mathbf{I} \tag{3.10}$$

$$\mathbf{J} = \int_0^\infty \mathbf{x}^T\mathbf{x}\mathrm{d}t = \mathbf{x}^T(0)\mathbf{P}\mathbf{x}(0) \tag{3.11}$$

The above formation handles the performance in term of tracking the setpoint ($\mathbf{x} = \mathbf{0}$). If the magnitude of the control signal should be counted, *i.e.*, the value of control signals **u** should be examined due to physical system bounds and/or control signal cost, the performance index can be rewritten as:

$$\mathbf{J} = \int_0^\infty (\mathbf{x}^T\mathbf{Q}\mathbf{x} + Ru^2)\mathrm{d}t \tag{3.12}$$

where R is the scalar weighting factor on the control signals. (Dorf, 1995)

This performance index is then minimized when:

$$\mathbf{K} = R^{-1}\mathbf{B}^T\mathbf{P} \tag{3.13}$$

and matrix **P** is determined from the solution of the equation:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}R^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{3.14}$$

Equation (3.14) is often called the Riccati equation, and this optimal control is called the Linear Quadratic Regulator (LQR).

LQR is often used for tracking some desired trajectory, *e.g.* set point temperature for an HVAC system. However, as the illustrated process suggested, this method would require the knowledge of matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$, which may not be available for some systems, *e.g.* the HVAC system and the plant/building thermodynamics.

Moreover, the performance index J provides no explicit constraints on the bounds of control signals **u**, making it difficult for situations where explicit bounds on control signals are posed *e.g.* cost limit constraint on HVAC operation. Indeed, in practice engineer needs to specify the weighting factors and compare the results with the specified design goals. Often this means that controller synthesis will still be an iterative process where the engineer judges the produced ”optimal” controllers through simulation and then adjusts the weighting factors to get a controller more in line with the specified design goals.

Furthermore, the LQR is not capable of addressing system noise, suppose an additive noise $\mathbf{v}(t)$ to the system:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{v}(t) \tag{3.15}$$

This noise gets accumulated with the integral of state variables $\mathbf{x}$, but not reflected in Equation (3.14), therefore the LQR would provide incorrect control signals. We provide a simple example in the following section to illustrate this issue.

3.1.3 System disturbance in LQR example

Suppose we have a system defined as:

$$\begin{cases} \dot{\mathbf{x}} & = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} & = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \end{cases} \tag{3.16}$$

where:

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} -2 & -1.5 & -1 & -0.5 \\ 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} & \mathbf{B} &= \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{C} &= \begin{bmatrix} 0.5 & 0.5 & 0.75 & 0.25 \end{bmatrix} & \mathbf{D} &= 0 \end{aligned} \tag{3.17}$$

The corresponding system transfer function for this system is:

$$G(s) = \frac{s^3 + 2s^2 + 3s + 1}{s^4 + 2s^3 + 3s^2 + 2s + 1} \tag{3.18}$$

We denote the coefficients in the numerator and denominator of this transfer function as:

$$num = \begin{bmatrix} 1 & 2 & 3 & 1 \end{bmatrix} \tag{3.19}$$

$$den = \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \end{bmatrix} \tag{3.20}$$

Suppose LQR is utilized to control the system with a state feedback **K**, with weighting factors in Equation (3.12) selected as:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, R = 10 \tag{3.21}$$

By solving Equation (3.13) and Equation (3.14), we get the state feedback gain:

$$\mathbf{K} = \begin{bmatrix} 0.4258 & 0.4665 & 0.4776 & 0.1531 \end{bmatrix} \tag{3.22}$$

Applying this state feedback $\mathbf{u} = -\mathbf{Kx}$ to the original system get a compensated system:

$$\begin{cases} \dot{\mathbf{x}} & = (\mathbf{A} - \mathbf{BK})\mathbf{x} \\ \mathbf{y} & = \mathbf{Cx} \end{cases} \tag{3.23}$$

The impulse response and step response of the original system and the compensated system are shown in Figure 3.1. One can observe that by introducing the feedback, the system will gain better performance on settle time and overshoot, and the steady state will drift to a new equilibrium point.

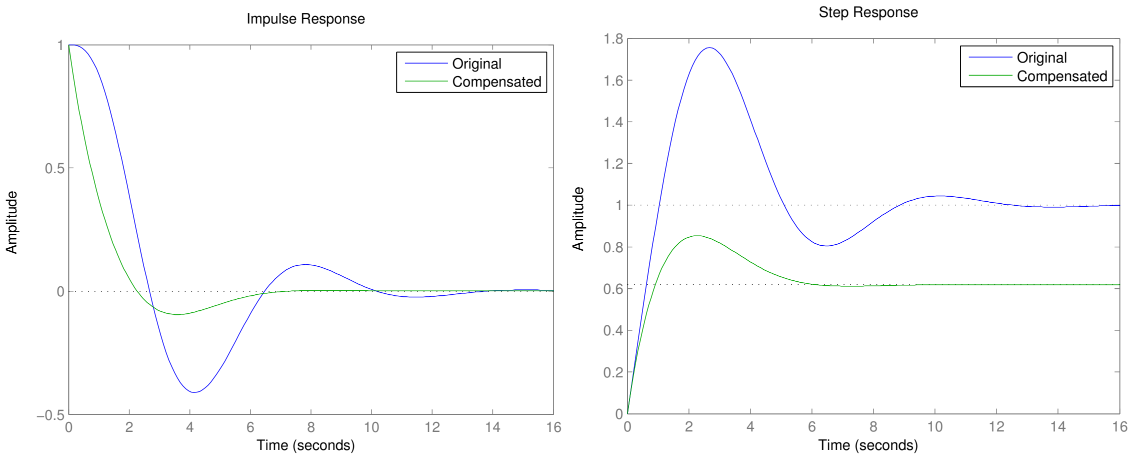
If a disturbance occurs on the system, indeed, suppose the system transfer function Equation (3.18) changes to:

$$G(s) = \frac{s^3 + 2s^2 + 3s + 1}{1.5s^4 + 2s^3 + 3s^2 + 2s + 1} \tag{3.24}$$

where a disturbance is introduced on the coefficient of term s^4 in the denominator.

The corresponding state space model of this disturbed system will be:

$$\mathbf{A}' = \begin{bmatrix} -1.33 & -1 & -0.67 & -0.67 \\ 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \end{bmatrix} \quad \mathbf{B}' = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.25}$$
$$\mathbf{C}' = \begin{bmatrix} 0.33 & 0.33 & 0.5 & 0.33 \end{bmatrix} \quad \mathbf{D}' = 0$$



(a) (b)

Figure 3.1: System response of original example system and state feedback compensated system. (a) Impulse response (b) Step response

With this disturbed system, if the feedback gain $\mathbf{K}$ is not updated, *i.e.* $\mathbf{K}$ remains the same as of Equation (3.22), applying it to the disturbed system yields:

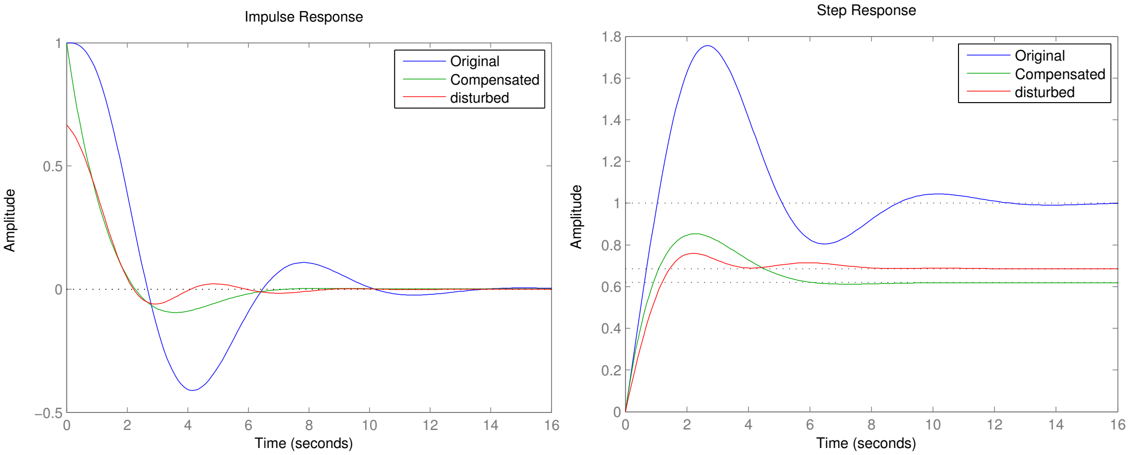
$$\begin{cases} \dot{\mathbf{x}} &= (\mathbf{A}' - \mathbf{B}'\mathbf{K})\mathbf{x} \\ \mathbf{y} &= \mathbf{C}'\mathbf{x} \end{cases} \tag{3.26}$$

Its impulse and step response is illustrated in Figure 3.3, as compared to the original system and compensated (undisturbed) system. The steady state again drifted to another equilibrium point.

Additionally, to ensure the control is optimal, one need to update the feedback gain $\mathbf{K}'$ for the disturbed system by solving Equation (3.13) and Equation (3.14), yielding:

$$\mathbf{K}' = \begin{bmatrix} 0.5587 & 0.4785 & 0.3696 & 0.1261 \end{bmatrix} \tag{3.27}$$

The above example shows that disturbances in the system will introduce errors on the output equilibrium point, as well as a need for updating the feedback gain. Figure 3.3a shows step response of 100 systems, generated by the example system described in Equation (3.18), with the coeffients in the numerator and denomintor



(a) (b)

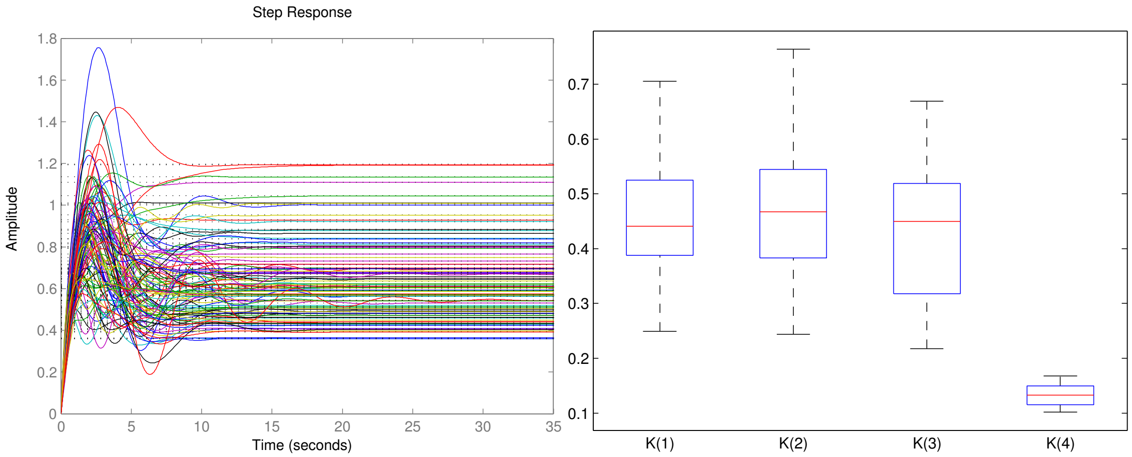
Figure 3.2: System response of disturbed system (a) Impulse response (b) Step response

disturbed by a uniform distributed noise $\sigma = U[0, 1]$. Figure 3.3b shows the box plot of the feedback gain values.

To address this problem, Linear Quadratic Gaussian control (LQG) was introduced (Athans, 1971; Speyer, 1979), where Kalman Filter (Kalman, 1960) and LQR are combined to address system uncertainty and/or incomplete state information. However, the LQG only address system uncertainty that is an additive Gaussian white noise, makes it unavailable for systems with non-Gaussian disturbance, *e.g.* HVAC operation will have system disturbances from outside temperature, in-house appliances, human activities, which are not necessarily Gaussian white noise distribution.

3.1.4 Model predictive control

Model Predictive Control (MPC), also called “receding horizon control”, is based on iterative, finite horizon optimization of a plant model. At time t the current plant state is sampled and a cost minimizing control strategy is computed for a relatively short time horizon in the future. Only the first step of the control strategy is



(a) (b)

Figure 3.3: System response and feedback gain of disturbed systems (a) Step response (b) Feedback gain values

implemented, then the plant state is sampled again and the calculations are repeated starting from the now current state, yielding a new control and new predicted state path (Camacho and Bordons, 2004). An illustration of the MPC process in discrete time is depicted in Figure 3.4.

For a plant modeled as $x(k+1) = f(x(k), u(k))$, the input within the optimization horizon N:

$$\mathbf{u}_{|k} = (u(k|k), u(k+1|k), ...u(k+N-1|k)) \tag{3.28}$$

is determined at each time k. The control input is calculated in order to minimize the predicted cost over the optimization horizon $k, k+1, ..., k+N-1$. And as described, the first control input $u(k|k)$ will be applied. (Camacho and Alba, 2013; Morari and H Lee, 1999)

MPC has advantages in handling complex system control with the built-in system model. Because its limited optimization horizon nature, it usually performs well in practice. However, it still requires the following to perform:

- an internal dynamic model of the process

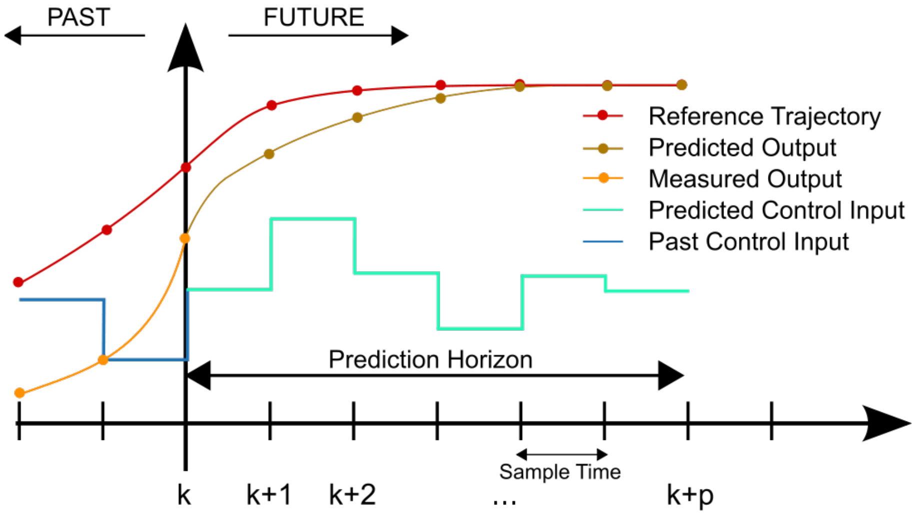


Figure 3.4: MPC scheme in discrete time

- a history of past control moves and
- an optimization cost function J over the receding prediction horizon

3.2 Data driven approximation

In this work we try to alleviate the requirement of building a plant/system model explicitly. Instead, we build an input-output relationship of the system from simple and linear approximation without specifying the system dynamics model, and this approximation is based on the data sampled, which may or may not be complete. In the following section we describe this work flow mathematically.

Suppose we have a target system with system state vector $\mathbf{x}$, system input/-control vector $\mathbf{u}$ and system output vector $\mathbf{y}$. The system dynamics is governed by a function $\dot{\mathbf{x}} = f(\mathbf{x})$ where $f(\cdot)$ can be either linear or non-linear. Moreover, assume we have some sensors that are capable of acquiring at least part of the state information and input/output information, *i.e.* the sampled state variables $\mathbf{x_s}$ is a subset of the system state variables: $\mathbf{x_s} \in \mathbf{x}$. For example, for the HVAC system described in previous chapters one of the system state variable, indoor temperature, was being sampled, while another variable, indoor humidity, was not. The same assumption holds true for sampled input, and output variables.

The task of the system approximation is then to find an estimation of the system input-output relationship:

$$\mathbf{\hat{y}_t} = \hat{f}(\mathbf{x_s}, \mathbf{u_s}) \tag{3.29}$$

where $\mathbf{\hat{y}_t}$ is the estimation of target output variable(s), and $\mathbf{x_s}$, $\mathbf{u_s}$ the partially sampled state and input variables respectively.

Moreover, we assume the target output variable(s) $\mathbf{y_t}$ is observable, meaning we can sample its value through sensors.

Assumption 1. *Target output vector* $\mathbf{y_t}$ *is a subset of observable output* $\mathbf{y_s}$*:*

$\mathbf{y_t} \in \mathbf{y_s} \in \mathbf{y}$

Ideally, we want to have the estimation function $\hat{f}(\cdot)$ to be a simple form to be used, such as first order linear function $\hat{f}(x) = \alpha_1 x_1 + \alpha_2 x_2 + \dots$. However, the system itself may not neccesarrily be a linear function relationship, therefore, some mapping functions may be needed to transform the orignal signals to another form. Concretely, in the proposed approach we assuem the estimation function a first order linear function as following:

$$\mathbf{\tilde{Y}_t} = \mathbf{A_0} + \mathbf{Az} \tag{3.30}$$

where $\mathbf{\tilde{Y}_t}$ is a diagonal matrix with:

$$\mathbf{\tilde{Y}_t} = \begin{bmatrix} \tilde{y}_1, 0, 0, \dots, 0 \\ 0, \tilde{y}_2, 0, \dots, 0 \\ 0, 0, \tilde{y}_3, \dots, 0 \\ \dots \end{bmatrix} \tag{3.31}$$

and $\tilde{y}_i$ is the mapping function of the target output variable:

$$\tilde{y}_i = \mathcal{F}(\hat{y}_{t_i}) \tag{3.32}$$

which maps the target output variable to another space $\tilde{y}$

The matrix $\mathbf{z}$ is defined as:

$$\mathbf{z} = \Big[\mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}, \dots\Big] \tag{3.33}$$

where $\mathbf{z_i}$ are augmented column vectors that are functions of sampled input and state variables:

$$\begin{cases} \mathbf{z_1} = & \mathcal{G}_1(\mathbf{x_s}, \mathbf{u_s}) \\ \mathbf{z_2} = & \mathcal{G}_2(\mathbf{x_s}, \mathbf{u_s}) \\ \dots & \end{cases} \tag{3.34}$$

with mapping functions $\mathbf{G}_i$

Finally the matrix $\mathbf{A}$ is the coefficient matrix:

$$\mathbf{A} = \begin{bmatrix} \alpha_{11}, \alpha_{12}, \alpha_{13}, \dots, \alpha_{1n} \\ \alpha_{21}, \alpha_{22}, \alpha_{23}, \dots, \alpha_{2n} \\ \alpha_{31}, \alpha_{32}, \alpha_{33}, \dots, \alpha_{3n} \\ \dots \\ \alpha_{n1}, \alpha_{n2}, \alpha_{n3}, \dots, \alpha_{nn} \end{bmatrix} \tag{3.35}$$

and $\mathbf{A_0}$ a constant offset vector.

Therefore, for a scalar-value mapped target output $\tilde{y}_1$, the above equation Equation (3.30) collapse to a one-dimensional situation:

$$\begin{aligned} \tilde{y} \quad &= \alpha_0 + \alpha\mathbf{z} \\ = \quad & \alpha_0 + \alpha_1 z_1 + \alpha_2 z_2 + \dots \end{aligned} \tag{3.36}$$

Indeed, the two mapping functions Equation (3.32) and Equation (3.34) essentially transform the original input and output vectors into another space such that relationship between the transformed output $\tilde{y}_i$ and input $\mathbf{z_i}$ can be approximated by a linear function Equation (3.30). And the goal for this problem becomes to find the transform functions in Equation (3.32) and Equation (3.34), such that the approximation generate the best fit in a least mean square sense.

Suppose the transform function is already known, there will be three possible outcomes from these transform functions, which is described in the following. For visualization purpose we will discuss it in a one-dimensional approximation scenario, *i.e.* $\tilde{y} = \alpha_0 + \alpha_1 z_1$, but the readers should be aware the this discussion also holds for higher dimensions.

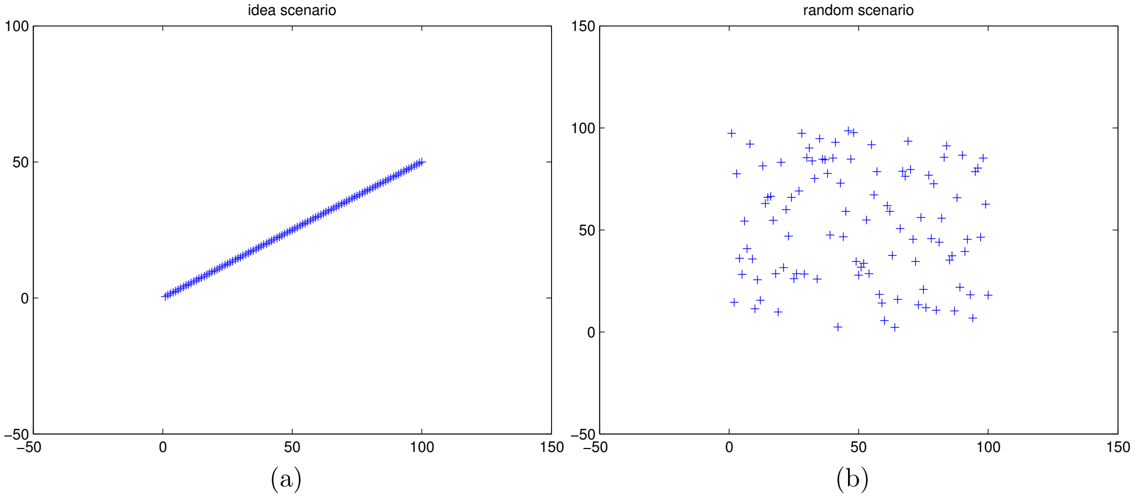


Figure 3.5: Simulation results comparing (a) Best case scenario: an idea case of a perfect linear relationship (b) Worst case scenario: a uniformly random distribution

1. Best Case Scenario: Ideally, if output $\tilde{y}_i$ and input $\mathbf{z_i}$ preserve a perfect linear relationship, as show in Figure 3.5a, then no further action needs to be taken, as the linear function describing their relationship can be utilized directly for the system approximation. However, this scenario rarely happens, if happens at all.

2. Worst Case Scenario:

   On the contrary, when the output $\tilde{y}_i$ and input $\mathbf{z_i}$ is statistically independent, or random as shown in Figure 3.5b, the previous approximation would fail in terms of capturing the main dynamics of the system under this scenario.

3. Desired Scenario

The desired scenario, which is the purpose of the transform functions, is to formulate the output $\tilde{y}_i$ and input $\mathbf{z_i}$ such that they show significant variance in one and only one direction, whereas insignificant variances in other directions. Concretely, let matrix $\mathbf{S_i}$ be a sample/observation recorded from $\tilde{y}_i$ and $\mathbf{z_i}$:

$$\mathbf{S_i} = \begin{bmatrix} \mathbf{z_i^{(1)}}^T, \widetilde{y_i^{(1)}} \\ \mathbf{z_i^{(2)}}^T, \widetilde{y_i^{(2)}} \\ \mathbf{z_i^{(3)}}^T, \widetilde{y_i^{(3)}} \\ \ldots \end{bmatrix} \tag{3.37}$$

where the upper script j in $\mathbf{z_i^{(j)}}$ means the jth sample.

The matrix $\mathbf{S_i}$ can be diagonalized by Singular Value Decomposition (SVD) as:

$$\mathbf{S_i} = \mathbf{U\Sigma V}^* \tag{3.38}$$

where $\mathbf{\Sigma}$ is a diagonal matrix whose value indicates the variances in each direction specified in the orthogonal basis $\mathbf{U}$, in a decreasing order. SVD would guarantee the first component in $\mathbf{\Sigma}$ is the most significant component of the data set. An example of SVD concept is shown in Figure 3.6b. Interested readers can refer to (Golub and Reinsch, 1970; Klema and Laub, 1980) for more information.

Therefore, the desired transformation should generate the matrix $\mathbf{S_i}$ such that its SVD $\mathbf{S_i} = \mathbf{U\Sigma V}^*$ preserve the following property:

$$\sigma_1 >> \sigma_j \quad j \neq 1 \tag{3.39}$$

where $\sigma_i, i = 1, 2, ...n$ is the diagonal values of Σ.

An example of such scenario is illustrated in Figure 3.6a

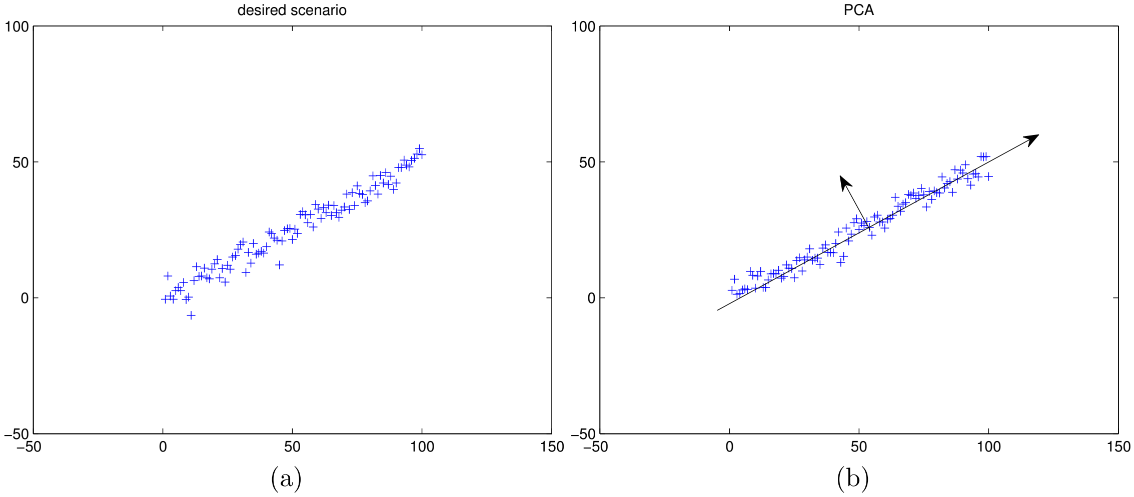


Figure 3.6: Simulation results comparing (a) Desired Scenario (b) Principle Component Analysis

3.3 Determining Transform Functions

As stated above, the transform functions play a key role in the process of approximation. However, as the targeted systems varies and their complex nature, it's infeasible to have a "general" rule for the transform function formulation that covers all possible systems. However, some approaches might be applied to tackle this problem depending on the problem setup. In the following section, we discuss some possible methods that can be helpful for some system characteristics in terms of formulating the transform functions.

1. System investigation/dynamics

   The first method that can be applied is to investigate the system dynamics, especially when the system dynamics is simple and obvious, such as first order systems. With these system the transform functions can be simply determined by the system dynamics.

   On the other hand, when the system dynamics is not as simple as a first order system, but may preserve some strong correlation between its input and output from investigation or observation, for example, the HVAC load and the outdoor temperature has a complicated relationship, but preserve a strong correlation that above a certain setpoint (cooling), the higher the outdoor temperature is, the higher HVAC load would be. It is usually feasible to formulate the transform functions based on the observation on their correlation relationship as well as some insights on the empirical data, through a set of simple mapping functions, including but not limited to:

- scaling: $f(x) = \alpha x$
- linear combination: $f(x, y) = \alpha x + \beta y$
- normalization: $f(x, y) = \alpha \frac{x}{y}$
- oscillation: $f(x) = \alpha \sin(x + \theta)$
- shifting: $f(x(t)) = x(t - \tau)$
- squaring: $f(x) = x^2$
- inversion: $f(x) = \frac{1}{x}$
- ...

Several mapping functions can also be combined to form a new mapping function. For example the HVAC load and outdoor temperature relationship discussed in this work utilized scaling, linear combination, as well as normalization to formulate the transform function.

2. Sample data pre-processing

Another technique could be utilized is to process the sampled data such that they come with a much simpler and concise form yet preserve the necessary information. Some common methods that could be applied to sampled data are:

- Normalization: $\mathbf{x}' = \frac{\mathbf{x} - \mathbf{E}(\mathbf{x})}{Max(\mathbf{x})}$

   - Averaging: $\mathbf{x}' = \mathbf{E}$
   - Offsetting: $\mathbf{x}' = \mathbf{x} + c$
   - Filtering: $\mathbf{x}' = \mathcal{F}(\mathbf{x})$
   - $\ldots$

   Where for filtering process different filters $\mathcal{F}(\cdot)$ can be applied for different purpose. One of the common application for applying a filter on sampled data is for outlier removal, which should be applied in case of sensor failure and/or malfunction.

3. Equilibrium

   Yet another possible approaches could be utilized is to distinguish different system behaviors, among which the one of most interest is the equilibrium points, if there were any. Usually equilibrium points means the system can be linearized and thus obtain a much simpler system description. Identifying equilibrium points are usually done by investigating the system dynamics, and/or visually examining the system outputs. After the equilibrium points are identified the system can be divided into different sections based on the these equilibrium point(s) and non-equilibrium point(s). Depending on the significance of these identified sections, some sections may be left out for system approximation if they are comparably insignificant to other sections.

It should be noted that the above mentioned approaches are by no means to be comprehensive, numerous other approaches and methods can be applied, based on a specific system setup. The following chapters shows this concept of data-driven approximation and optimization utilized on the HVAC system problem described in Section 2.3.

CHAPTER 4

Prediction Model

As depicted in Figure 2.1 the proposed approach includes a prediction engine and an optimization engine. In this chapter the development of prediction engine is discussed. We start with the model description of the thermostat as well as the building, followed by the modeling and approximation schemes for the indoor temperature and the HVAC load. We also point out a "low load" problem associated with HVAC system when the average usage time of running HVAC system is low *e.g.* heating load in winter in low latitude locations like Arizona, and discuss a solution for this issue.

4.1 Models

4.1.1 Smart Thermostat Model

The thermostat is responsible for HVAC control and indicates if the HVAC unit should be "on" or "off" through the control signal, u. HVAC activity is based on a comparison of the set point function for a given time, and the current temperature. Although the mode of the thermostat can be set to heat or cool, for simplicity the following description assumes the system is in cool mode, with the corresponding inequalities to be reversed for heat mode.

The set point for the controller is denoted as r, and the internal temperature of the home is denoted as x. A simple thermostat (in cooling mode) with cycle state can then be defined with the following behavior:

$$u(t) = \begin{cases} 1 & \rho > y(t) \wedge \neg \text{cycle} \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

This simple thermostat is state based, where *cycle* is set to true for some time after $u(t)$ switches from 1 to 0. This prevents the thermostat from switching on immediately after it switches off. Cycle time is a necessary component of an HVAC unit, since the internal physical components must rest periodically for efficiency reasons.

An alternative formalism expresses the thermostat using a hysteresis is given as:

$$u(t) = \begin{cases} 1 & x(t) > r(t) + \epsilon \\ 0 & x(t) < r(t) - \epsilon \\ u(t^{-}) & otherwise \end{cases} \tag{4.2}$$

where $u(t^{-})$ represents the previous value of $u(t)$. If ϵ is sufficiently large, then the cycle time will be an emergent property of the system (based on the behavior of the building (plant), and the actual weather conditions).

Using the formalism of (4.2) makes for easier analysis of the system since as a discrete mode is not needed for system state. However, pathological external factors could still reduce the cycle time, resulting in chattering. However, we will assume that using a value such as $\epsilon = 1$°F will prevent chattering with external temperatures up to 130°F. This formalism has no dependency on how $r(t)$ is specified, and operates merely on the incoming values of $x(t)$ and $r(t)$ for the specified time.

For a traditional programmable thermostat, the set point function is piecewise constant and can be defined as:

$$r(t) = \begin{cases} r_0 & t_0 \leq t < t_1 \\ r_1 & t_1 \leq t < t_2 \\ r_2 & t_2 \leq t < t_3 \\ r_3 & t_3 \leq t < \bar{t} \end{cases} \tag{4.3}$$

where $t \in [0, \bar{t})$, and $\bar{t}$ represents the end of the day. As a shorthand, we can rewrite this as

$$r(t) = \{(r_0, t_0), (r_1, t_1), (r_2, t_2), (r_3, t_3)\} \tag{4.4}$$

where the series repeats. The values of t for such a thermostat are obtained from considering only the hours of the day, where the successor for $\bar{t}$ is $t_0 = 0$. Modes such as “Morning”, “Work”, etc., can be placed in these ranges.

In this work we propose a similar piecewise constant thermostat, but without a requirement that the series repeat daily, with no fixed times at which set points can change, and no constraint on the number of changes per day. An optimization algorithm may choose to constrain number of changes per day, but that choice is not constrained by a physical device. Then the time-varying setpoints are then stated as

$$r(t) = \{(r_0, t_0), (r_1, t_1), \ldots\} \tag{4.5}$$

where r_n and t_n may be chosen by the external optimization engine.

In a similar fashion, a 5/2 Programmable thermostat would have sets of values of $y(t)$, depending on whether the day was a weekday ($d \leq 5$) or weekend ($d > 5$), therefore has the name 5/2, meaning five weekdays and two weekend days.

$$y(t) = \begin{cases} y_0 & (t_0 \leq \tilde{t} < t_1) \wedge d \leq 5 \\ y_1 & (t_1 \leq \tilde{t} < t_2) \wedge d \leq 5 \\ y_2 & (t_2 \leq \tilde{t} < t_3) \wedge d \leq 5 \\ y_3 & (t_3 \leq \tilde{t} \vee t < t_0) \wedge d \leq 5 \\ y_4 & (t_0 \leq \tilde{t} < t_1) \wedge d > 5 \\ y_5 & (t_1 \leq \tilde{t} < t_2) \wedge d > 5 \\ y_6 & (t_2 \leq \tilde{t} < t_3) \wedge d > 5 \\ y_7 & (t_3 \leq \tilde{t} \vee t < t_0) \wedge d > 5 \end{cases} \tag{4.6}$$

For sake of brevity, the case for the 7-day programmable thermostat is not shown, though it is clearly a generalization of the function of $y(t)$ shown in (4.6).

4.1.2 Building Model

The internal temperature of a home is denoted as $x(t)$, outside temperature as $d(t)$, and internal temperature set point as $r(t)$. HVAC utilization over a particular time interval $[t, t+\tau)$ is abbreviated as $w(t, \tau)$. In discrete form, the internal temperature of the home can be expressed as

$$x(t + \tau) = x(t) + \Delta x(t, \tau) \tag{4.7}$$

where the function $\Delta x(t, \tau)$ represents changes to internal temperature brought about by external energy, and HVAC system usage. $\Delta x(t, \tau)$ can be expanded as follows:

$$\Delta x(t, \tau) = \Delta x_d(t, \tau) + \Delta x_w(t, \tau) + \sigma(t) \tag{4.8}$$

where $\Delta x_d(t, \tau)$ is the change due to external energy over time τ, $\Delta x_w(t, \tau)$ is the change due to HVAC system usage over time τ, and $\sigma(\cdot)$ is the disturbance attributed to the system or the prediction process.

$$\begin{aligned} \Delta x_d(t, \tau) &= g(x(t), d(t), \tau) \\ \Delta x_w(t, \tau) &= f(x(t), d(t), r(t), \tau) \end{aligned} \tag{4.9}$$

The energy used by the HVAC system can be directly calculated over a horizon (if $\Delta x_w(t, \tau)$ is known) as follows:

$$w(t, N) = \Sigma_{i=0}^{N-1} \Delta x_w(t + i\tau, \tau) \tag{4.10}$$

4.1.3 Model Imperfection

It is noted that Equation (4.8) and Equation (4.10) exclude several terms required to model HVAC load to arbitrary precision. In (Daou et al., 2006) the HVAC load is a combination of the temperature (sensible load) as well as humidity (latent load). Other disturbances such as temperature of the equipment, occupants and their activities, etc., also factor into the HVAC load.

Concretely, the design cooling load (or heat gain) is the amount of heat energy to be removed from a house by the HVAC equipment to maintain the house at indoor design temperature, and there are two types of cooling loads: sensible cooling load, and latent cooling load.

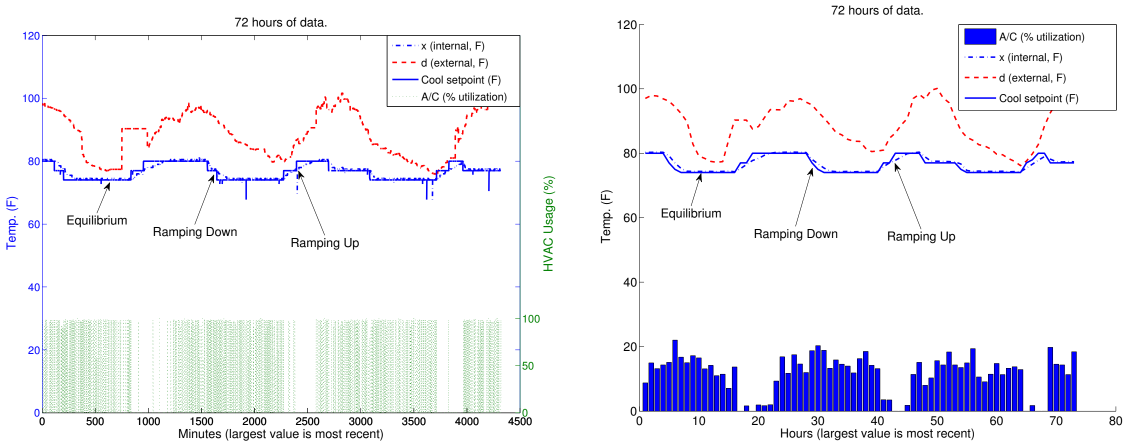
The sensible cooling load refers to the dry bulb temperature of the building and the latent cooling load refers to the wet bulb temperature of the building. In the summer, humidity influence in the selection of the HVAC equipment and the latent load as well as the sensible load combined will affect the HVAC operational load.

The model we build ignores some of these terms since obtaining data to account for these values would require a user to install and invest in equipment whose value may not be recouped by the savings gained. Since many common thermostats have only temperature information, we implicitly assume that the latent load is consistent (or at least periodic) during the 7-10 day horizon we are observing, so it becomes a lumped parameter in our model.

Even with these generous assumptions, the model described in Equation (4.8) and Equation (4.10) is an appropriate approximation based on the limited (inexpensive) sensors commonly available. In the following sections we demonstrate the feasibility of the proposed method.

4.2 Load Prediction

Based on the proposed thermostat and building models, a data-driven approach is employed to construct a prediction model in four phases, namely data pre-processing, approximation, regression, and prediction. In the data pre-processing phase, sampled raw data is filtered and used to determine linear approximations of Δx_d and Δx_w. The purpose of this phase is to prepare the raw data, remove any outliers and/or spikes that might be introduced by the sensor, and smooth out the data for the next phase. In the approximation phase, a technique called Averaged Sectional Model (ASM) is developed to capture the main characteristics of the sampled data, by dividing the data into different sections based on their thermal behavior. Simplification techniques such as regression is then performed in the regression phase to build a closed-form linear relationship to estimate usage as calculated in (4.10). Finally in the prediction phase, the approximated model is used to calculate an estimation of future HVAC usage, and periodic recalculation of this functions is performed to ensure the system is able to respond to deviations in the expected weather patterns. The following sections detail the steps needed to perform prediction of the HVAC utilization.



(a) (b)

Figure 4.1: (a) Raw data (72 Hours) obtained from a single-family Arizona home in July 2011 (b) Hourly Averaged raw data of subreffig:aveRawData

4.2.1 Data Pre-processing

Data are collected from three single-family homes and one office space in Arizona as the testbed for this work. Sampling is performed by a commercial off-the-shelf wireless thermostat with an indoor temperature sensor and HVAC monitoring capability. Each sample obtained from the thermostat contains six data fields: indoor temperature (x), outdoor temperature (d), HVAC cooling set point (r^c), HVAC cooling usage (w^c), HVAC heating set point (r^h), HVAC heating usage (w^h). For brevity, only cooling set points are discussed, and we set $r = r^c$ and $w = w^c$ for the rest of the discussion. The sampled temperatures and set points are in Fahrenheit while the unit for HVAC usage is in percentage, specifically 0% indicates the HVAC is OFF while 100% indicates the HVAC ON. Figure 4.1 shows a horizon of three days of raw and hourly-averaged data obtained from one of the residential homes in July 2011. Information pertinent to heating ($r^h$and w^h) are omitted from the figure.

From the data obtained (Figure 4.1) several observations can be made. Firstly, the indoor temperature correlates closely to the cooling set point (r), as is the goal of an HVAC system. Secondly, in most instances of HVAC utilization the HVAC is either (nearly) fully ON (98%) or (nearly) fully OFF (10%). Upon brief analysis, one can observe that the HVAC runs for a set period of time to “drop” the indoor temperature at/below the cooling set point, then turns off until the temperature reaches a threshold at/above the cooling set point before repeating the cycle again. Thirdly, a few outliers exist in the raw data, due to sensor inaccuracy and/or missed sensor reading due to a request time out. Lastly, the time series of indoor temperature and set points can be roughly divided into three main categories, namely, Equilibrium, Ramp-up and Ramp-down. Equilibrium points are where the indoor temperature closely correlates to the set point, *i.e.* the indoor temperature is “clapped” to the set point because of HVAC operation. Ramp-up correspond to areas where the set point changes to a higher temperature and results in near zero HVAC utilization (in cooling mode). In these areas the indoor temperature will slowly drift to the new higher set point, based on the temperature difference between indoor and outdoor temperature. On the other hand, the Ramp-down periods reflect time periods where the set point is lowered. Accordingly, the HVAC utilization nears 100% utilization as the HVAC system attempts to quickly meet this new set point value. Each of these categories can be observed in Figure 4.1.

Based on these observations two major simplification techniques, namely sectioning and averaging, are applied to the raw data to facilitate the data analysis process, as described below.

The sectioning procedure aims at classifying the different categories of the data (Equilibrium, Ramp-up, Ramp-down) and throw away insignificant samples that might be too short in time. This process is two-fold, (i) raw data is divided into regions, based on x and r, (ii) within each region if sufficient consecutive samples exist before the region changes, the data is utilized for further considered, otherwise this data is removed. Due to the intrinsic difference between Equilibrium points and Ramp-up/Ramp-down points, the corresponding simplification techniques are slightly different. For Equilibrium points, data samples for determining $w(\cdot)$ come from regions where $|x - r| < T^*$, indicating the internal temperature samples are in a "uniform temperature region" (*i.e.* equilibrium), being held there by HVAC utilization while disturbed by outside temperature. At least N^* consecutive samples must exist within the region to be utilized. On the other hand, the Ramp-up/Ramp-down samples for determining $w(\cdot)$ come from regions where $|x-r| > T^*$, indicating the internal temperature samples have "drifted away" from the specified set points. Specifically, $x-r > T^*$, indicating the Ramp-down points and $r-x > T^*$ indicating the Ramp-up points. This process is depicted in Figure 4.2.

After obtaining valid regions, an averaging process is then applied on these regions. For uniform regions, the average of all data fields (x,d,r,w) within that region producing an Averaged Sectional Model for the equilibrium points. For example, a uniform region is identified from 4900 to 4970 (min) with an average HVAC usage of 13.03%, average indoor temperature of 80.43°F, average outdoor temperature of 77.44°F and and average cooling set point of 80°F. After scanning all equilibrium regions, the same process is again applied to all the Ramp-up and Ramp-down regions to determine their corresponding Averaged Sectional Model. The obtained Averaged Sectional Model is also depicted in Figure 4.2.

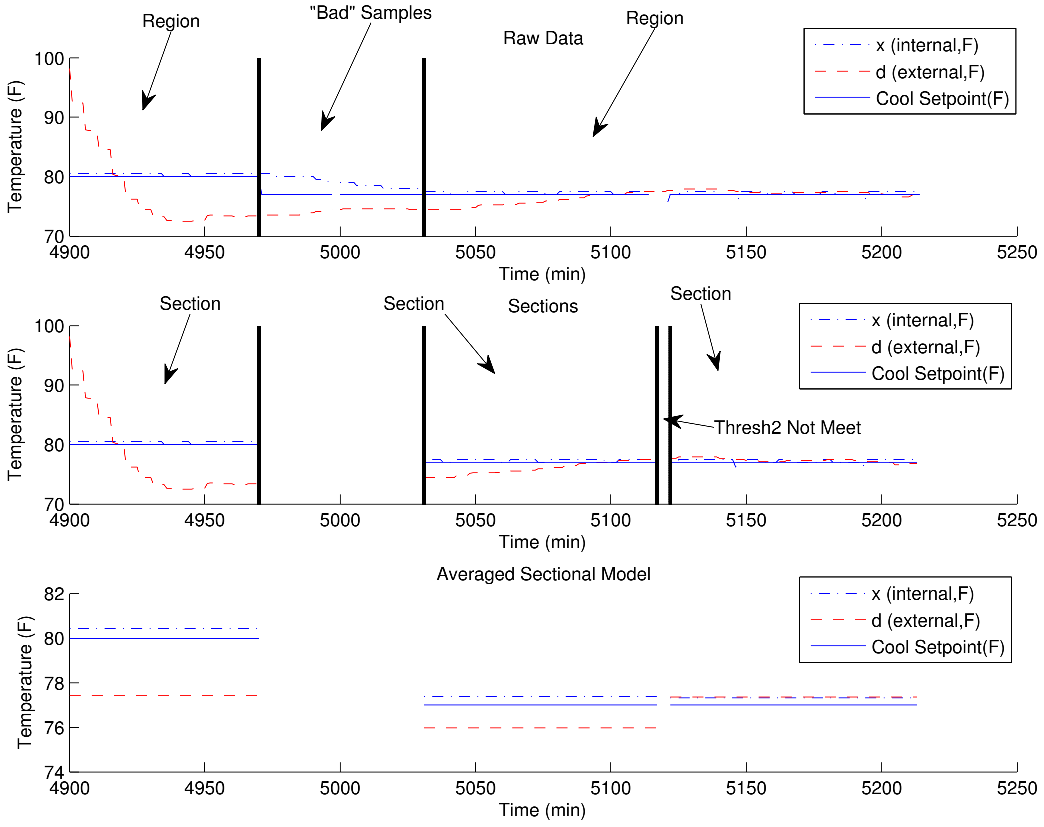


Figure 4.2: Averaged Sectional Model for Equilibrium sections

It should be noted that by performing the averaging techniques, the outliers resides in the raw data can be effectively removed and smoothed out.

By using this method, the prediction engine is now able to translate the discrete on/off nature of the HVAC usage into a more meaningful percentage, while maintaining a finer-grained approximation of the relationship between outdoor temperature and the HVAC utilization required to maintain a given set point. Moreover, by utilizing the averaging technique there is a reduction in the “bumping” effect of HVAC usage due to the cycling ON and OFF periods, thus produce a quantitative metric of HVAC usage of a section.

4.2.2 Approximation Schemes

Following the initial data pre-processing is the approximation phase, where a linear relationship between temperature and HVAC utilization for the three categories are being established, which forms the foundation of the prediction engine in the proposed framework. In the following sections we describe in details the methodology of building up this approximation scheme for each section category.

Equilibrium Section

Recall from (4.8) that changes in temperature is a function of the changes due to external energy (Δx_d) and HVAC system usage over time (Δx_d) as well as disturbance, σ. If the system is in equilibrium (where $|x(t) - r(t)| < \epsilon$ for all $t \in [t_0, t_1]$), and the disturbance is taken to be Gaussian, we can take the sum of the temperature changes over the span $[t_0, t_1]$, with $t_1 - t_0 = N\tau$, and N is a positive integer:

$$\begin{aligned} \Sigma_{i=0}^{N-1} \Delta x(t_0 + i\tau, \tau) &= \Sigma_{i=0}^{N-1} \Delta x_d(t_0 + i\tau, \tau) \\ &\quad + \Sigma_{i=0}^{N-1} \Delta x_w(t_0 + i\tau, \tau) + 0 \end{aligned} \tag{4.11}$$

where the sum of the (Gaussian) disturbances of the time frame can be approximated as zero. In addition, if the system is in an uniform temperature region, we can say that $x(t_0) \approx x(t_1)$, which implies that $\Sigma_{i=0}^{N-1} \Delta x(t_0 + i\tau, \tau) \approx 0$, i.e.,

$$\Sigma_{i=0}^{N-1} \Delta x(t_0 + i\tau, \tau) \approx 0 = \Sigma_{i=0}^{N-1} \Delta x_d(t_0 + i\tau, \tau) + \Sigma_{i=0}^{N-1} \Delta x_w(t_0 + i\tau, \tau) \tag{4.12}$$

We approximate the change in temperature (usually an increase) over a short time, τ, as a linear relationship between the difference in the outside and inside temperature with $\alpha \in \mathbb{R}$:

$$\Delta x_d(t_0, \tau) = \alpha(d(t_0) - x(t_0))\tau \tag{4.13}$$

A corresponding *decrease* in temperature (over a short time) occurs when the HVAC system is active, namely using the following approximation with $\beta \in \mathbb{R}$:

$$\Delta x_w(t_0,\tau) = \begin{cases} -\beta(d(t_0) - x(t_0))\tau & x(t_0) > r(t_0) + \epsilon \\ 0 & x(t_0) < r(t_0) - \epsilon \\ \Delta x_w(t_0 - \tau, \tau) & otherwise \end{cases} \tag{4.14}$$

The discontinuous nature of Δx_w makes analysis difficult to do in closed form. Fortunately, the knowledge that Δx_w may be "off" for one or more time spans of length τ permits the addition of a utilization function during equilibrium (EQ) $\tilde{w}^{EQ}$:

$$\Delta x_w(t_0,\tau) = \tilde{w}^{EQ}(d(t_0), r(t_0))\left(-\beta(d(t_0) - x(t_0))\tau\right) \tag{4.15}$$

where $\tilde{w}^{EQ} \in [0,1]$ relates how often the HVAC must be "on" based on the outside and set point temperatures at t_0. Typically this utilization function is upper bounded by physical equipment constraints, such as required time for a compressor to rest between executions. Rewriting (4.12):

$$\begin{aligned} \Sigma_{i=0}^{N-1}\Delta x_d(t_0 + i\tau, \tau) &= -\Sigma_{i=0}^{N-1}\Delta x_w(t_0 + i\tau, \tau) \\ \Sigma_{i=0}^{N-1}\alpha(d(t_0 + i\tau) - x(t_0 + i\tau))\tau &= \Sigma_{i=0}^{N-1}\tilde{w}^{EQ}(d(t_0 + i\tau), r(t_0 + i\tau))\beta(d(t_0 + i\tau) - x(t_0 + i\tau))\tau \end{aligned} \tag{4.16}$$

This formula is also clearly difficult to analyze in closed form, so a convenient approximation is not to consider the time indexed values of x, d, r, but rather the average differences between them. Thus, let $t^* = N^{-1}\Sigma_{i=0}^{N-1}(d(t_0 + i\tau) - x(t_0 + i\tau))$, and assume (given the uniform temperature region that puts the system in equilibrium) that $r(t) = x(t) \forall t \in [t_0, t_1]$. Since t^* is a constant, we have the following:

$$\begin{aligned} \alpha t^* &= \beta t^* \Sigma_{i=0}^{N-1}\tilde{w}^{EQ}(d(t_0 + i\tau), x(t_0 + i\tau)) \\ \alpha &= \beta \Sigma_{i=0}^{N-1}\tilde{w}^{EQ}(d(t_0 + i\tau), x(t_0 + i\tau)) \end{aligned} \tag{4.17}$$

One final linear approximation defines a model for $\tilde{w}^{EQ}$ with $\gamma \in \mathbb{R}$:

$$\tilde{w}^{EQ}(d(t_0), r(t_0)) = \gamma(d(t_0) - r(t_0)) = \gamma(d(t_0) - x(t_0)) \tag{4.18}$$

using $r(t) = x(t)$, this give the following approximation of utilization:

$$\tilde{w}^{EQ}(d(t_0), r(t_0)) = \gamma t^* \tag{4.19}$$

Thus, for a given t^* a utilization is expected, and the pair are linearly related. As stated previously, this approach has utilized the assumption of uniform temperature region (where $r(t) = x(t)$), so utilization definitions are valid *only* for a specific set point, meaning that for each $r(t)$ value, a new linear relationship must be defined. Fortunately r is typically set at integer values between 60 and 85 (for Fahrenheit thermostats), so the number of required linearization is not excessive.

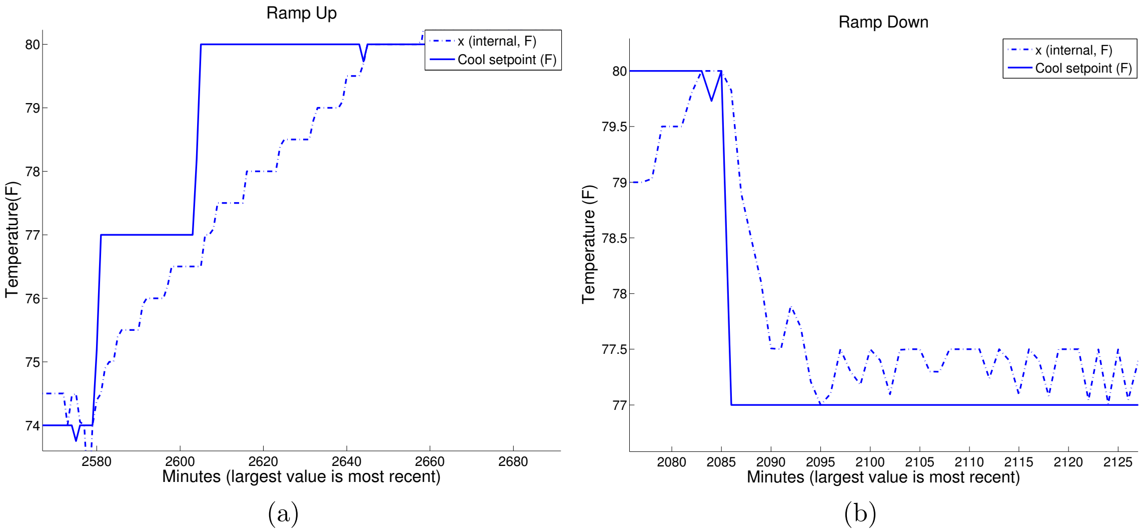


Figure 4.3: Illustration of ramp sections (a) Ramp Up section (b) Ramp Down section

Ramp Regions

The Ramping regions have different thermal behaviors as compared to the Equilibrium regions. Figure 4.3a provides a Ramp-up example. The set point was increased, yielding a period of little to no HVAC utilization as the indoor temperature increases due to the higher outdoor temperature. In this region, the HVAC prediction engine is conceded with the time it takes for the temperature to “float” to the higher value.

From (4.8) the change in temperature is a function of the changes due to external energy and HVAC system usage over time (4.9) as well as disturbance, σ. When the system is in a Ramp-up region the HVAC utilization per unit time T_0 is a constant $\tilde{w}^{RU_{T_0}} = \tilde{w}_0^{RU_{T_0}}$, ideally $\tilde{w}_0^{RU_{T_0}} = 0$, and this can be calculated out from averaging the HVAC usage over the Ramp-up region. What is of interest is the transition time T^{RU} required. Similar to the analysis of Equilibrium sections, we can take the sum of the temperature changes over the span $[t_0, t_0 + T^{RU}]$, with $T^{RU} = \Sigma_i \tau(i)$, $\tau(i) \in \mathbb{R}, i \in \mathbb{N}^+$ as:

$$\Sigma_{i=0}\Delta x(t_0 + i\tau(i), \tau(i)) \quad = \quad \Sigma_{i=0}\Delta x_d(t_0 + i\tau(i), \tau(i))$$

Approximating the time $\tau(i)$ needed for the indoor temperature $x(i)$ to transition to the new setpoint $r(i)$ based on the current outdoor temperature $d(i)$ yields:

$$\tau(i) = \theta^{RU}((d(i) - x(i)), (r(i) - x(i))) \tag{4.20}$$

A similar analysis can be utilized for Ramp-down regions, except the HVAC would operate near full capacity to decrease the indoor temperature to the new set point. Figure 4.3b provides an example of Ramping-down. Again, the HVAC utilization would be a constant $\tilde{w}^{RD_{T_0}} = \tilde{w}_0^{RD_{T_0}}$, theoretically $\tilde{w}_0^{RD} = 100\%$, and the approximation of transition could be determined by:

$$\tau(i) = \theta^{RD}((d(i) - x(i)), (r(i) - x(i))) \tag{4.21}$$

Thus, based on this analysis the HVAC usage given ramping up and ramping down regions can be determined as follows:

$$\tilde{w}^{RU} = \tilde{w}^{RU_{T_0}} \times \tau(i) = \tilde{w}^{RU_{T_0}} \times \theta^{RU}(d(i), r(i), x(i)) \tag{4.22}$$

$$\tilde{w}^{RD} = \tilde{w}^{RD_{T_0}} \times \tau(i) = \tilde{w}^{RD_{T_0}} \times \theta^{RD}(d(i), r(i), x(i)) \tag{4.23}$$

4.2.3 Data Regression

The next step is then to build a relatively accurate model for HVAC usage estimation, based on the sampled data. The approaches taken in this work is through regression, as described in the following.

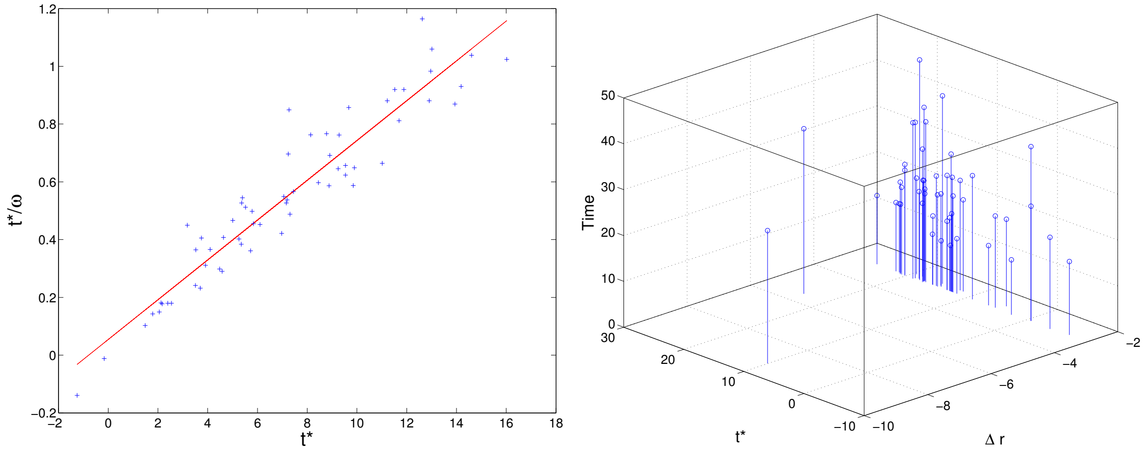
Equilibrium Regions

For equilibrium sections the HVAC utilization function $\tilde{w}^{EQ}$ needs to be estimated. For this purpose, the averaged data for each section (denoted by i) from Averaged Sectional Model discussed in Section 4.2.1 is grouped (denoted by j) based on their set point r. Figure 4.4a illustrates the relationship of w^i_j and t^{*i}_{j} where $r_j = 74°\text{F}$ for data sampled in July, 2011. Note that each point in the graph represents a section i. The x-axis denotes t^{*i}_{j} while the y-axis is w^i_j normalized by t^{*i}_{j}, denoted as w^i_{Nj}. This plot provides an intuitive impression that the relationship of w^i_j and t^{*i}_{j} is quasi-linear, indicating the possibility of utilizing a linear regression within the regression kernel. The regression method is applied to each group. The normalized usage w_N is used, thus for group j the estimation would be:

$$\tilde{w}^i_{Nj} = \tilde{f}_j(t^{*i}_{j}) \tag{4.24}$$

We denote the target normalized HVAC usage of group j as a column vector $\mathbf{w}_{Nj} = [w^1_{Nj}, w^2_{Nj}..., w^m_{Nj}]^T$, m is the number of sections in group j, and the corresponding estimation a column vector $\tilde{\mathbf{w}}_{Nj} = [\tilde{w}^1_{Nj}, \tilde{w}^2_{Nj}, ..., \tilde{w}^m_{Nj}]$. The purpose of regression is then to find a suitable function $\tilde{f}_j$ that minimize $\|\tilde{\mathbf{w}}_{Nj} - \mathbf{w}_{Nj}\|_2$ for all groups $\forall j \in G$.

The HVAC prediction engine similarly employs linear regression to formulate the estimation function $\tilde{f}_j$. Specifically, $\tilde{\mathbf{w}}_{Nj} = \tilde{f}_j(t^*{}_j, a) = X_j\theta_j$, where X_j is the $m \times n$ training data matrix for group j is constructed as follows:



(a) Data Regression

(b) Ramp sections

Figure 4.4: Raw data selected from a 30 day period, when $r = 74$°F. (a) Linear regression of equilibrium point. (b) Estimation of time to complete ramp region.

$$X_j = \begin{array}{c} \\ s_1 \\ s_2 \\ \dots \\ s_m \end{array} \begin{array}{c} \begin{array}{cccc} ft_1 & ft_2 & \dots & ft_n \end{array} \\ \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{pmatrix} \end{array}$$

where each row represents an individual section (s_i) and the columns reflect the features (ft_j) of each section. The first column of each row is set to one to represent the bias offset. The minimization problem then becomes: $\arg\min_{\theta_j} \|X_j\theta_j - \mathbf{w}_{Nj}\|_2$ where θ_j is a $n \times 1$ vector indicating the regression parameters. After solving the minimization problem the estimation of a single section can be calculated as:

$$\tilde{w}^i_{Nj} = X^i\theta_j = ft^i_{j1}\theta_{j1} + ft^i_{j2}\theta_{j2} + \dots + ft^i_{jn}\theta_{jn} \tag{4.25}$$

Within this work, the training data is chosen as $ft^i_{j1} = 1$, $ft^i_{j2} = t^{*i}$, and $others_j = 0$, i.e. X_j will be a $m \times 2$ matrix. Figure 4.4a shows the linear regression result.

Ramping Regions

Conversely, in the Ramp-up/down sections the HVAC utilization is constant. Thus, the HVAC Prediction Engine is responsible for the estimation of the transition time to either ”float” to a higher set point or “drop” the temperature to a lower set point. A similar approach is taken for these ramping sections, where relationship between transition time $T^{R\{U,D\}}$ and t^* are considered for each set point group. Figure 4.4b shows the relationship for $r_j = 74°F$. Similar to the equilibrium sections, the ramping section samples also exhibit quasi-linear properties. Thus a linear regression method can also be utilized to to estimate the transition time:

$$T_j^{R\{U,D\}} = \tilde{\theta}_j({t^*}^{i}_{j}) \tag{4.26}$$

The HVAC utilization estimation for the ramping sections is a linear relationship based on the estimated transition time T^{R*} and the section HVAC utilization constant $\tilde{w}_0^{R*}$:

$$\tilde{w}^{R\{U,D\}} = g(T^{R\{U,D\}}, \tilde{w}_0^{R\{U,D\}}) \tag{4.27}$$

Outlier detection

It could happen that the ASM data points may have some outliers too which could represent unusual or extreme weather conditions, or other abnormal situations such as humidity change etc. In this case the linear regression might get affected if running directly with these outliers in the data set. Therefore, outlier detection and removal procedure should be deployed. In this work, the outlier removal is conducted by a Principle Component Analysis (PCA) procedure followed by a statistical process. The PCA procedure identifies the principle component or principle axis in the data while the statistical process removes any data points that is “far away” from this axis. Readers could refer to (Abdi and Williams, 2010; Jolliffe, 2005) for more details

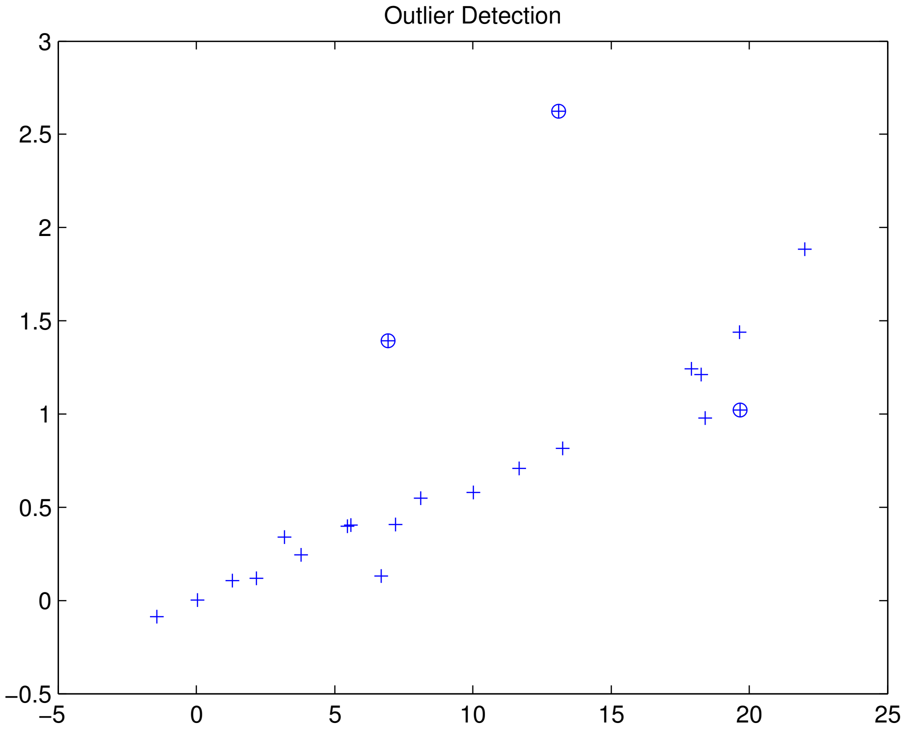


Figure 4.5: Outlier removal

in PCA.

Concretely, the distance of the test data point to the principle axis is utilized to determine whether this point is an outlier or not. This is done by calculating the Euclidean distance defined as:

$$\sqrt{\Sigma_{i=1}^{n}(\mathbf{x_i}-\mathbf{y_i})^2} \tag{4.28}$$

where i indicates the different component directions, $\mathbf{x_i}$ is the tested point and $\mathbf{y_i}$ represents the mean of the sampled data.

An outlier is then identified if the defined Euclidean distance is above a certain threshold after normalization.

An example of this process is shown in Figure 4.5 where the ASM data points are plotted as "+", where outliers in these data points are detected and indicted by an overlaid "o".

Artificial Disturbance

To account for natural disturbances such as changes in the outdoor temperature or human activity within the building, an artificial disturbance is incorporated within the regression result. The artificial disturbances is in the form of a zero mean Gaussian white noise, which is additive to the estimation function:

$$\tilde{w}_{Nj}^{EQ_i} = \tilde{f}_j({t^*}^i_j, a_j) + gwn(0, \sigma) \tag{4.29}$$

$$\tilde{w}_{Nj}^{R\{U,D\}_i} = g(\tilde{\theta}_j({t^*}^i_j, a_j), \tilde{w}_0^{R\{U,D\}}) + gwn(0, \sigma) \tag{4.30}$$

where $gwn(0, \sigma)$ is a zero mean Gaussian white noise with variance= σ^2. It is to be noted that the periodicity and daily load profile are already integrated in this formation by differentiating the various set point temperature groups j, and by grouping the data point within the same set point temperature, therefore the set of discrete approximations would account for different load profile situations. To improve this model of occupant behavior would require additional information or sensors, so our work makes this somewhat coarse approximation. However, the results show that the approximation is useful, as we show next.

4.2.4 Prediction Formulation

Given (4.29), HVAC usage can be predicted in Equilibrium regions if the difference between outdoor temperature and the set point temperature is know. The outdoor temperature can be obtained from weather forecasting services. The linear regression produces normalized estimation functions $\tilde{f}_j$ for each group $j \in G$. The estimation of future outdoor temperature $d(t) = d(nT)$ is obtained from a weather forecasting service, and the indoor temperature set point $r(t) = r(nT)$ is obtained from the user. For $n \in N$ and T in the sample time frame, the normalized estimation function $\tilde{f}_j$ is utilized where group j is chosen to have the same set point temperature as input:

$r_j = r(n)$. Next, the output is denormalized with respect to $t^*{}_j(n)$ to obtain the percentage of HVAC usage $\tilde{w}_j(n)$ at sample n.

Similarly, given (4.30), HVAC usage can be predicted for Ramping regions if the difference between outdoor temperature and the set point temperature is known, as well as the based on the nominal HVAC usage for the transitions $\tilde{w}_0^{R\{U,D\}}$ for each group. As in the Equilibrium regions, the linear regression for Ramping regions will produce estimation functions $\tilde{\theta}_j$ for each group $j \in G$ and provides an estimated transition time, as well as the nominal HVAC usage $\tilde{w}_{0j}^{R\{U,D\}}$ for that group. By utilizing the weather forecasting information and the user defined set point value, equation(4.30) is able to predict HVAC usage for the ramping regions.

4.2.5 Adaptation

Overtime the characteristics of the outdoor-set point relationship may be affected by external changes such as changes in season, abnormal weather patterns, among others. These disturbances will lead to inaccuracy, therefore the estimation functions $\tilde{f}_j$,$\tilde{\theta}_j$ adapt as the observed data obtained changes. Two regression update policies are utilized, namely a fixed update rate policy and an error threshold update policy.

The fixed update rate policy mandates that the regression is conducted at a fixed time interval τ_{up} specified by the algorithm. This method would ensure the regression is updated every τ_{up} time, to ensure the estimation functions continue to reflect the underlying system. Conversely, the error threshold update policy only triggers the regression update if an error between the predicted HVAC usage at time t and the observed HVAC usage is greater than a predefined error threshold $\tilde{T}$:

$$\tilde{e} = |\tilde{w}(t) - w(t)| > \tilde{T} \tag{4.31}$$

These policies work together to ensure the estimation functions reflect the underlying system while avoiding unnecessary regression updates.

4.3 Low Load Problem

The previous stated approach for HVAC load prediction is able to obtain a reasonable accuracy (90%) when the HVAC load demand is high. For example, the HVAC cooling load in the hot summer in Arizona, or heating load in a cold winter in Minnesota. However, when the demand of HVAC operation is sparse, the above approach may face a challenge of estimating an accurate HVAC load. For example, the HVAC heating load in the winter in Arizona is approximately 12%, and temporally sparse. This is basically due to the fact that the indoor temperature is almost in between the user set heating and cooling points, therefore no HVAC operation is needed. This artifact will affect the previous approaches severely because the sparse nature will lead less or even no training data for the approach to do the regression, therefore an inaccurate model might be developed, and the following prediction would be incorrect. We call this problem a “low load problem”, where the load factor of HVAC operation is low.

To overcome this problem, a revised framework for HVAC load prediction is developed.

4.4 Load Prediction Revised

The revised HVAC load prediction scheme consists one more component, namely Indoor Temperature Prediction engine, to provide some insights on the possible HVAC load factor. Concretely, this component would first make an estimation on the future indoor temperatures, where this estimation is feed to the HVAC load prediction engine. With this new information, the HVAC load prediction engine would then revise the prediction result based on the estimated indoor temperature. Moreover, a new type of section denoted as “Free section” will be recognized along with Equilibrium, Ramp Up and Ramp Down sections. In Free section, the estimated indoor temperature fully comply with the setpoint temperature schedules, *i.e.* the estimated indoor temperature is below the cooling setpoint temperature and above the heating setpoint temperature, therefore no HVAC operature is needed in these Free sections.

Generally, the revised approach is comprised of two phases. In the first phase the indoor temperature of the target building is predicted using an ARMAX model, based on weather forecasting, user defined setpoint temperatures and previous indoor temperatures. In the second phase the heating/cooling load is predicted based on the predicted indoor temperature, and linearized to create the HVAC prediction model based on the three sections as described in Averaged Sectional Model, excluding the Free sections. The details steps are described in the following.

4.4.1 Indoor Temperature Prediction

In the first phase an Auto Regressive Moving Average with eXogenous inputs (ARMAX) model is used to predict the indoor temperature. The predicted indoor temperature would then provide a reference for HVAC load prediction in a later stage.

A discrete time linear ARMAX model is based on a simple linear difference equation:

$$
\begin{aligned}
y(t) = & -(a_1 y(t-1) + ... + a_{n_a} y(t-n_a)) \\
& + b_1 u(t-n_k) + ... + b_{n_b} u(t-n_k-n_b+1) \\
& + c_1 e(t-1) + ... + c_{n_c} e(t-n_c) + e(t) \\
= & -\Sigma_{n=1}^{n_a} a_n y(t-n) + \Sigma_{m=1}^{n_b} b_m u(t-n_k-n_b+1) \\
& + \Sigma_{q=1}^{n_c} c_q e(t-q) + e(t)
\end{aligned}
$$

where $y(t)$ is the estimated output (indoor temperature), t denotes time, a_n, b_m, c_q are unknown system parameters, $u(t)$ denotes the input, which could be a matrix when multiple input signals are chosen, n_a, n_b, n_c are orders associated with output $y(t)$ and inputs, n_k is the number of input samples that occur before the input affects the output, also called the *dead time* in the system and $e(t)$ is unknown system error.

In this work the indoor temperature is estimated and serves as the model output, while the model inputs are outdoor temperatures (composed of sampled outdoor temperature history data as well as weather forecasting information) as well as the user defined setpoint temperatures.

The order of the ARMAX inputs, i.e. the value of n_a, n_b, n_k, n_c, should be determined using the parsimony principle, "out of two or more competing models which all explain the data well, the model with the smallest number of independent parameters should be chosen" (Söderstrom and Stoica, 1989).

In this work, a range of values for n_a, n_b, n_k, n_c were evaluated, with each value ranging from 1 to 5. Each combination of the system parameters was estimated using the MATLAB System Identification Toolbox.Data used for the system training and verification process are obtained from the first 70% of each month's sampled data. The system model that results in the best model fit would then be used to represent the model.

4.4.2 Load Prediction

After the indoor temperature prediction phase, the same techniques discussed in Section 4.2 will be used to establish the ASM for the building. However, the load prediction formula would be slightly different from previous ones, where two alternatives to the prediction formula can be used, namely model modification and post-process modification.

Model Modification

The model modification approach involves modifying the regression model for HVAC load prediction. More specifically, a new input, the estimated indoor temperature, is added to the regression model, therefore factoring in the new information.

Based on this approach, the regression model described in Section 4.2.3 will have some minor modifications. Concretely, the regression model Equation (4.32) will have the indoor temperature estimation $\hat{x}$ as another input, and rewritten as:

$$\tilde{w}^{i}_{Nj} = \tilde{f}_j(t^{*i}_{\;j}, \hat{x}) \tag{4.32}$$

where $\hat{x}$ can also be a linear combination of indoor temperature and setpoint temperature r, such as $\hat{x} - r$, etc.

Post-process Modification

Another way to factor in the indoor temperature estimation is through a method called post-process modification. Intuitively, the effect of indoor temperature on the HVAC load follows an "indicator" fashion, which can be described a set of logic as following:

$$\begin{cases} \text{If } x > r^h \;\&\; x < r^c : & \text{HVAC} = 0 \\ \text{If } x < r^h : & \text{HVAC} = 1 \\ \text{If } x > r^c : & \text{HVAC} = 1 \end{cases} \tag{4.33}$$

where

$$\text{HVAC} = \begin{cases} 0 & \text{:HVAC on} \\ 1 & \text{:HVAC off} \end{cases} \tag{4.34}$$

indicating the operation status (ON/OFF) of the HVAC system, based on the indoor temperature.

Thus HVAC load prediction can be obtained by first calculating the original linear regression model in Section 4.2.4, then apply the above logic to perform prediction modifications.

Because of the discrete and binary nature of this indicator effect, the post-process modification is a better solution for generating the HVAC load prediction than the model modification approach, since the former takes similar binary logic while the later still requires a linear model.

Load Prediction Process

The final load prediction with indoor temperature estimation then can be formulated as the work flow depicted in Figure 4.6.

Given a prediction horizon, the indoor temperature of that horizon is first estimated using the ARMAX model described in Section 4.4.1. Secondly, based on the indoor temperature prediction, coupled with the user defined target set point temperature, the four classes of sections can be extracted. Free sections are first identified by comparing the estimated indoor temperature and the target setpoint temperature. Next, the start time of Ramp-Up and Ramp-Down sections are identified by scanning for changes in the setpoint temperature. The end time of these ramping sections however are unknown and need to be estimated. Next, the HVAC usage prediction for Equilibrium sections is carried out using Equation (4.29) as a baseline estimation, with free section HVAC usage set to zero. Finally the transition time of ramps sections are estimated using Equation (4.30), and the baseline estimation is refined by the nominal HVAC usage in these transition regions.

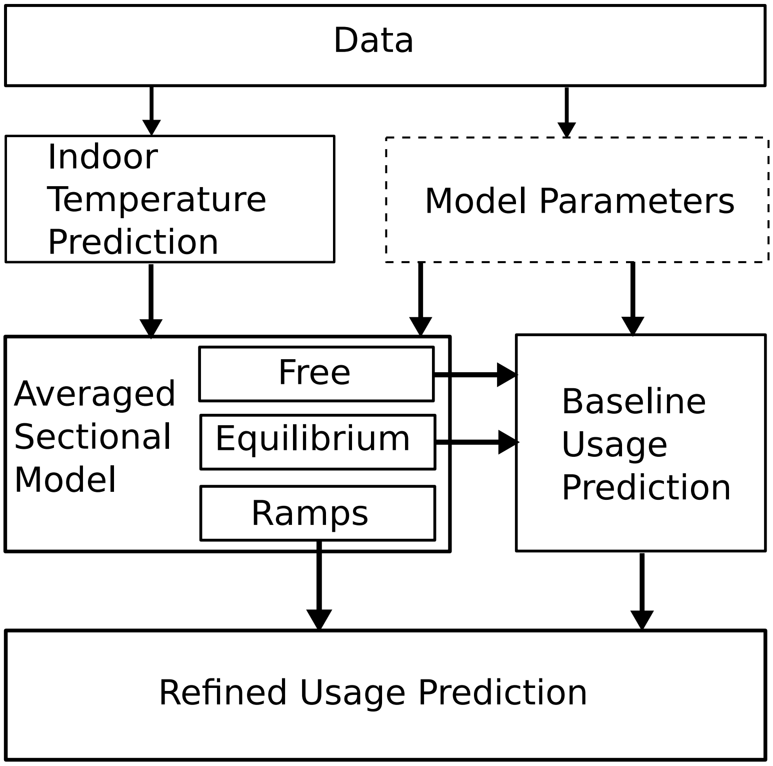


Figure 4.6: HVAC load prediction work flow

CHAPTER 5

Optimization Engine

The HVAC prediction engine described in the previous chapter enables users with the ability to inject a variety of set points to discern the resulting HVAC energy and cost, providing users with accessible metrics to understand the long-term consequences of these decision. Alternatively, the HVAC prediction engine can be coupled with an optimization engine which enables an end user to specify a cost constraint (i.e., a cost setpoint), and produce the corresponding optimized schedule which adheres to the user defined cost constraint, while striving to minimize the difference between the actual and desired user defined internal set point. By integrating an optimization component within the framework, a closed loop system is created in which users no longer have to determine the set point schedule. Also, the framework can adapt to the environment and take advantage of time-of-use pricing policies (Tucson Electric Power (TEP), 2012b) biasing HVAC utilization to non-peak times, and perform temperature-cost trade-offs. Moreover, as changes in the platform or predicted weather are detected, the optimization component can dynamically adapt the set point schedule.

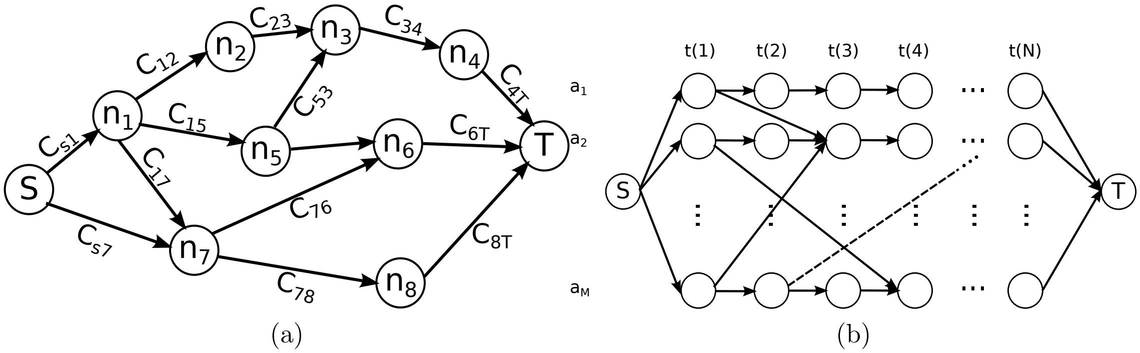
In this chapter, we discuss the approach and implementation of such an optimization engine, based on a slight modification of classic minimal cost network flow problem. In addition, we illustrate solving this problem by using a "off-the-shelf" optimization package by providing an example.

5.1 Approach

5.1.1 Problem Formulation

As described previously, the optimization problem can be restated as finding the optimal schedule of set point temperatures $S(t)$ over a time horizon $t \in (t_0, t+T]$ such that the optimized set points are close to the user-specified set points, while meeting the user-specified cost constraint. Minimizing a cost function given user constraints can be reduced to a linear programming problem. While many algorithms exist for efficiently solving this class of problems (Ahuja et al., 2003), we formulate the set point scheduling problem as a modified minimum cost network flow problem (Figure 5.1), with the details stated as following.

Minimum Cost Network Flow



(a) (b)

Figure 5.1: (a) Typical Network Flow Model (b) Illustration of Constructed Network Flow Model for Cost Limited Setpoint Optimization

Figure 5.1a illustrates a typical network flow model setup for solving a minimum cost flow problem. The formulation of this problem can be stated as following (Ahuja et al., 2003): Let $G = (N, A)$ be a directed network defined by a set N of n nodes and a set A of m directed arcs. Each $arc(i, j) \in A$ has an associated cost C_{ij} that denotes the cost per unit flow on that arc. We associate with each $arc(i, j) \in A$ a capacity U_{ij} that denotes the maximum amount that can flow on the arc and a lower bound l_{ij} that denotes the minimum amount that must flow on the arc. We associate with each node $i \in N$ an integer number $b(i)$ representing its supply/demand. If $b(i) > 0$, node i is a supply node; if $b(i) < 0$, node i is a demand node with a demand of $-b(i)$; and if $b(i) = 0$, node i is a transshipment node. The decision variables in the minimum cost flow problem are arc flows and we represent the flow on an $arc(i, j) \in A$ by x_{ij}. The typical minimum flow problem then would be formulated as:

$$Minimize \sum_{(i,j)\in A} C_{ij}x_{ij} \tag{5.1a}$$

subject to

$$\sum_{\{j:(i,j)\in A\}} x_{ij} - \sum_{\{j:(i,j)\in A\}} x_{ji} = b(i), \ \text{for all}i \in N \tag{5.1b}$$

$$l_{ij} \leq x_{ij} \leq U_{ij}, \ \text{for all}(i, j) \in A \tag{5.1c}$$

The constraints in Equation (5.1b) is referred as mass balance constraints, where the first term represents the $outflow$ of the node i and the second term represents the $inflow$. If the node is a supply node (denoted as S) then $b(i) > 0$ would hold true, if the node is a demand node (denoted as T) then $b(i) < 0$ would hold true, otherwise the node would be a transshipment node where the outflow equals inflow, i.e. $b(i) = 0$. The amount of flow on $arc(i, j)$ should also satisfy the minimum and maximum limit of that arc, which is referred as flow bound constraints as in Equation (5.1c).

Modified Minimum Cost Network Flow

Next, we will illustrate how to formulate the cost constrained setpoint optimization problem as a modified minimum flow problem, by first structuring the nodes and arcs of the network, defining the cost functions, and adding additional constraints.

Consider a node matrix A with $M \times N$ nodes, as shown in Figure 5.1b, where M represents the total number of all available setpoints at each decision point (controlled by the optimization granularity, equals to the sampling rate by default), the value of each node in one column are denoted as a_i and it's duplicated for every column. N represents the total length of time scope for optimization, i.e. a node $\{A_{ij}, (i \in M, j \in N)\}$ represents a setpoint node at time j with value a_i. On the other hand, an $arc(A_{i_1 j_1}, A_{i_2 j_2})$ linking two nodes $A_{i_1 j_1}$ and $A_{i_2 j_2}$ represents a setpoint transition, i.e. setpoint changed from a_{i_1} at time j_1 to a_{i_2} at time time j_2. If $a_{i_1} = a_{i_2}$ the setpoint value is not changed, we call this an equilibrium arc, if$a_{i_1} > a_{i_2}$ or $a_{i_1} < a_{i_2}$ the setpoint value is shifted to a new setpoint, we call them ramp-down or ramp-up arcs respectively. Since the ramp sections would take some non-zero time for the indoor temperature to transit from the current setpoint to new setpoint, as stated in Section 4.2.2, when constructing these arcs the previous-stated ramp analysis techniques would be employed to estimate the time needed for the setpoint value transition, which is then reflected in the network by connecting the corresponding nodes in the right column. For example, in Figure 5.1b at $t(1)$, if the setpoint would transit from a_1 to a_2 it would take 2 time units (hours in this example), therefore the arc would connect the node A_{11} with node A_{32} in the third column, rather than A_{22} in the second column. Moreover, we put one constraints $\{arc(A_{i_1 j_1}, A_{i_2 j_2}) : j_1 \neq j_2\}$ on the arcs based on the assumption that only one setpoint can be set to at one time, therefore no arcs exist between nodes within the same column. The mass of flow on each arc is the variable that we will optimize over, and is an indicator variable x_{ij} defined as,

$$x_{ij} = \begin{cases} 1 & select\ arc(i,j) \\ 0 & otherwise \end{cases} \quad (i,j) \in A \tag{5.2}$$

The goal of the optimization is to provide the users a setpoint schedule that is close to their desired or preferred temperature while keep the cost below the specified limit. Therefore we define the *distance* between the user-preferred setpoint $Desired_j$ at time j and the actual selected (by optimization engine) setpoint Set_{ij} on node A_{ij} as $D_{ij} = abs(Desired_j - Set_{ij})$. The monetary cost on the other hand, is associated with arcs, where the cost of an $arc(i, j)$ between node i, j is defined as the the predicted cost of HVAC usage C_{ij} and obtained from the previous prediction engine as in Equation (4.29) and/or Equation (4.30) in Section 4.2.4, which can be rewritten as C_{ij} is a function of the outside temperature x, setpoint r and Gaussian White Noise.

$$C_{ij} = g(u(j,t)) = g(f(Set(t,j), wT(t,j), gwn(0,\sigma))) \tag{5.3}$$

Another constraint that is introduced to the optimization engine is the number of transitions constraint, i.e. how many times at maximum that the optimizer can reset the setpoint to a new value that is different from it's previous value. This constraint is introduced to emulate the real-world setpoint schedule strategy from users where the user would set the temperature to a new point only a few times a day, for example in the "morning","working" and "sleeping" scheme the setpoints only change between these three different time span. And this constraint would avoid the artifacts that the optimization may result in a "bumping" schedule where the setpoints change too frequently. The number of transitions $tran$ is calculated as following:

$$tran = \sum_{(i,j)\in A} x_{ij} transit(i,j) \tag{5.4}$$

$transit()$ is a function to check if the setpoint value of node i is different from node j:

$$transit(i,j) = \begin{cases} 0 & value(i) = value(j) \\ 1 & otherwise \end{cases} \quad (i,j) \in A \tag{5.5}$$

and $value(i)$ is just the setpoint value associated with node i:

$$value(A_{mn}) = a(m), A_{mn} \in A \tag{5.6}$$

With the components, the setpoint optimization problem can be formulated as the following,

$$Minimize \sum_{(i,j)\in A} D_{ij}x_{ij} \tag{5.7a}$$

subject to

$$\sum_{(i,j)\in A} C_{ij}x_{ij} \leq B \tag{5.7b}$$

$$tran = \sum_{(i,j)\in A} x_{ij}transit(i,j) \leq TR \tag{5.7c}$$

$$\sum_{\{j:(i,j)\in A\}} x_{ij} - \sum_{\{j:(i,j)\in A\}} x_{ji} = b(i), \text{ for all}i \in N \tag{5.7d}$$

$$l_{ij} \leq x_{ij} \leq U_{ij}, \text{ for all}(i,j) \in A \tag{5.7e}$$

Where $D_{ij}x_{ij}$ generates the temperature difference (penalty) and $C_{ij}x_{ij}$ generates the cost of selecting $arc(i,j)$. B specifies the end user defined cost constraint. Equation (5.7b) ensures the monetary cost of the optimized setpoint would be within the user specified limit. TR in Equation (5.7c) is the maximum number allowed for setpoint transitions. The algorithm used to generate the network is depicted in Algorithm 1.

The optimization task can be described as follows: at each time step t, new sensor readings and up-to-date monetary cost as well as number of transitions are gathered and utilized to update the corresponding constraints B and TR. The new sensor reading may utilized by the prediction engine to update the linear regression process as stated in Section 4.2.5. Then all nodes within a given time scope are considered and the corresponding arcs are generated by Algorithm 1. For each node and the arcs the distance and cost values for are determined, as well as the number of the transitions.

Algorithm 1 Network Flow Model Generation

```
for i=1 to n do
  for j=1 to m do
    calculate $D_{ij}$ for node $A_{ij}$
    for k=1 to m do
      if $value(A_{i,j}) == value(A_{i+1,k})$ then
        create $arc(A_{i,j}, A_{i+1,k})$
        calculate $C_{A_{i,j},A_{i+1,k}}$ for $arc(A_{i,j}, A_{i+1,k})$
      else
        estimate ramp time $t$
        create $arc(A_{i,j}, A_{i+floor(t),k})$
        calculate $C_{A_{i,j},A_{i+floor(t),k}}$ for $arc(A_{i,j}, A_{i+floor(t),k})$
      end if
    end for
  end for
end for
```

5.1.2 Solving Minimum Cost Network Flow, by Example

In this work, we utilize the CPLEX tool from IBM to solve the network flow problem. The CPLEX is a powerful tool for solving mathematical programming problems including linear programming, mixed integer linear programming, quadratic programming, and mixed integer quadratic programming etc. A quick example of using CPLEX is as following:
Suppose the problem to be solved is

$$\text{Maximize } x_1 + 2x_2 + 3x_3 + x_4 \tag{5.8a}$$

$$\text{subject to}$$

$$-x_1 + x_2 + x_3 + 10x_4 \leq 20 \tag{5.8b}$$

$$x_1 - 3x_2 + x_3 \leq 30 \tag{5.8c}$$

$$x_2 - 3.5x_4 = 0 \tag{5.8d}$$

$$\text{Bounds}$$

$$0 \leq x_1 \leq 40 \tag{5.8e}$$

$$0 \leq x_2 \tag{5.8f}$$

$$0 \leq x_3 \tag{5.8g}$$

$$2 \leq x_4 \leq 3 \tag{5.8h}$$

$$\text{Integers} \tag{5.8i}$$

$$x_4 \tag{5.8j}$$

The corresponding CPLEX model can be created and solved in MATLAB as in Listing 5.1:

```
% Initialize the CPLEX object
cplex = Cplex('mipex1');
cplex.Model.sense = 'maximize';


% Use addRows to populate model
```

```
cplex.addCols([1; 2; 3; 1], [], [0; 0; 0; 2], [40; inf; inf; 3],'CCCI
  ');
cplex.addRows(-inf, [-1  1 1    10], 20);
cplex.addRows(-inf, [ 1 -3 1     0], 30);
cplex.addRows(   0, [ 0  1 0  -3.5],  0);

% Optimize the problem
cplex.solve();
```

Listing 5.1: CPLEX code example

The three *addRows* lines can be merged into one line by populating the constraints in the matrix format, specifically, we can define

```
lhs = [-inf;-inf;0];
rhs = [20;30;0];
A = [-1,1,1,10;1,-3,1,0;0,1,0,-3.5];
```

Listing 5.2: CPLEX code example: Alternative constranit matrix

then the constraints can be rewritten in one line as:

```
cplex.addRows(lhs,A,rhs);
```

Listing 5.3: CPLEX code example: Alternative constranit

Example Network

CPLEX can be used for the HVAC network flow problem described in our work. For the sake of simplicity, we use a simple network flow constructed in Figure 5.2 as an example. Firstly, we will first show how to model the target function and the cost constraint as well as the balance function from this model, then we will show how to add the transition constraint by adding augmented variables to the model such that it is in compliance with the network flow model described in Equation (5.7).

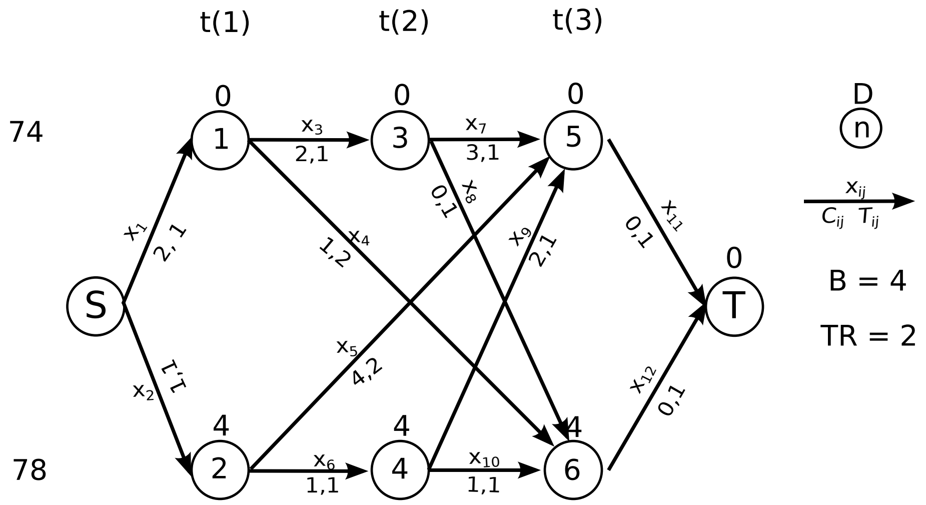


Figure 5.2: A simple network example

Suppose the network is constructed as in Figure 5.2, where at each time step there are only two possible set point candidates, $74F$ and $78F$, and there are total three time steps $t(1)$, $t(2)$ and $t(3)$. The nodes are numbered in a sequential way from $node(1)$ to $node(6)$, together with a source node $node(S)$ and a sink node $node(T)$. The user wants to keep the temperature to $Desired_{ij} = 74F$ for all the time $i, j \in A$, therefore the associated *distance* or D_{ij} can be calculated as $D_{ij} = abs(Desired_{ij} - Set_{ij})$, which is annotated on each node. The capital cost of choosing each arc is computed as C_{ij} and the time each arc needs is computed as T_{ij}. For a better descriptive purpose we reformat the variables x_{ij} into a column vector x_i. Suppose we are constructing the arc from $node(1)$, which is set to $74F$, to point to a new set point at $78F$, assume we have calculated the capital cost of this arc will be 1 dollar and the time needed for the transition is 2 hours, therefore the arc is constructed as connecting $node(1)$ at $t(1)$ to $node(6)$ at $t(3)$, as depicted in the figure. Moreover, we denote the number of nodes in each time layer as m, the total number of nodes in the network as M, and the number of arcs as N, in this example $m = 2$, $M = 8$ and $N = 12$

First let's consider the target function. If we define the decision variables x_i

associated with each arc as a column binary vector $\mathbf{x}$, the calculated distance for each arc D_i is essentially the distance of the destination node of the arc, for example, the distance of $arc(4)$ is $D_4 = D(node(6)) = 4$. Since the sink node is only for the purpose of balancing the network flow, the distance on the sink node would always be zero. These distances are defined as a row vector $\mathbf{D}$. The target function can be simply put as

$$min\ \mathbf{Dx} \tag{5.9}$$

where in this example $\mathbf{x} = [x_1, x_2, ...x_{12}]^T$, and $\mathbf{D} = [0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 0]$

Next, the capital cost constrain inequality and the balance function are considered. Assume the capital cost of each arc is calculated as in Figure 5.2, the total budget is B, then we can simply construct the capital cost as a row vector $\mathbf{C}$ and compose the capital cost constrain as

$$\mathbf{Cx} \leq B \tag{5.10}$$

where in this example $\mathbf{C} = [2, 1, 2, 1, 4, 1, 3, 0, 2, 1, 0, 0]$ and $B = 4$. Again, the cost of the last two arcs pointing to the sink node would always be zeros.

The balance function would constrain the flow balance on each node, to ensure the inflow and outflow are balanced for all the nodes except the source node and the sink node, i.e. the number of inflow and the number of outflow should be equal for all the transition nodes. Since we need to have a balance function for each node as shown in Equation (5.7d), there will be the number of nodes balance functions in total, to put them into the matrix format this would be $\mathbf{A_{eq}x} = b$, where $\mathbf{A_{eq}}$ is a $M \times N$ matrix and $\mathbf{x}$ is the previous stated N column vector. To construct this $\mathbf{A_{eq}}$ we need to know the inflow arcs and outflow arcs of each node, or in another way, the source and destination node of each arc. This is where the time delay information is required. The arcs are constructed in this network by using Algorithm 1. When constructing the arcs, the algorithm begins sequentially on each node, namely the source node $node(S)$, then $node(1)$, $node(2)$,...,till the sink node $node(T)$ is reached.

On each node, the algorithm will create m arcs flowing out of the node into each one of the candidate setpoints nodes, where the horizontal location of the candidate node is determined by the calculated time delay. For example, assume we are creating the arcs for $node(1)$ at $t(1)$ in the figure, first we create the arc flowing into the first candidate node which is a node with $value = 74F$, the calculated time delay is 1 so we find the node in the next time layer $t(2)$ as $node(3)$, then we create the arc flowing into a node with $value = 78F$, the calculated time delay is now 2, therefore we find the node $node(6)$ in $t(3)$ as the destination node. Because each node will have precisely m arcs flowing out, and these arcs are constructed in a sequential way of traversing all the candidate temperatures, it's then easy to track each arc's source and destination nodes, therefore in turn to calculate the inflow and outflow arcs of each node. If we denote a outflow as a negative flow while inflow as a positive flow, the $\mathbf{A_{eq}}$ in the example can be calculated as:

$$\mathbf{A_{eq}} = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{5.11}$$

The above matrix can be easily validated by investigating each column and check if there is one and only one 1 as well as one and only one -1 on each column. The intuition behind this is the fact each arc must be connecting from one node to another, where the former node sees this arc as an outflow (-1) and the later sees it as an inflow (1).

On the other hand, all the nodes expect the source node S and the sink node T are transition nodes, therefore the net flow on those nodes should be zero. Source node S will have a net flow -1 while the sink node has 1. Thus the net flow vector

$$\mathbf{b} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.12}$$

And the balance function could be written as

$$\mathbf{A_{eq}x} = \mathbf{b} \tag{5.13}$$

At this point we have constructed the target function as well as the capital cost, the other constraint needs to be constructed is the number of transitions constraint which limit the total number of setpoint changes. To construct this constraint, we will need to introduce some augmented variables to the current solution.

First, we need define what is a transition in the constructed network. A transition is a change of setpoint temperature in time. For example, $arc(1,6)$ means setpoint temperature changes from 74F to 78F, thus counted as a transition. What we need to find out is the number of transitions happened. If we investigate all the outflow arcs for each node, it is clear that each node would have m outflow arcs and $m-1$ of them would be counted as a transition, the remaining one is the one going to the same setpoint temperature at the next time step. If we can find a way of representing whether a transition is taken at each node, we can easily figure out the total number of transitions of the network by summing each node's transition up. By recognizing this properly, we can create an augmented variable for each node to represent if this node would result a transition for each inflow arc. Suppose on node n one of the inflow arcs is $arc(i)$ with flow x_i, there are m outflow arcs, $arc(j_1)$, $arc(j_2)$,...,$arc(j_i)$,...,$arc(j_m)$, where $arc(j_i)$ points to the same setpoint temperature as the source node of inflow $arc(i)$, and the flow on the outflow arcs are x_{j_1}, x_{j_2},..., respectivel. Then we can create a variable x_{ai}:

$$x_{ai} = x_i - x_{j_i} \tag{5.14}$$

to represent if a transition happens at node n for inflow from $arc(i)$. For example, in Figure 5.2, the augmented variable for $node(1)$ with inflow x_1 will be $x_{a1} = x_1 - x_3$.

If the flow on x_1 satisfies $x_1 = 1$, meaning the setpoint temperature on $node(1)$ is chosen to be 74F, and the outflow on $x_3 = 1$, meaning the next setpoint temperature is chosen to be 74F, there will be no transition, and $x_{a1} = x_1 - x_3 = 0$. If the next setpoint temperature is chosen to be 78F, then we have $x_3 = 0$ and $x_4 = 1$, thus $x_{a1} = 1$, there will be one transition.

If we do the same calculation on for each inflow arc on the node and for each of the nodes in the network, we will have the number of transitions taken at each node, where the total number of transitions with an augmented variable $x_a = \sum_i x_{ai}$.

Now we extend our variable vector $\mathbf{x}$ with these augmented variable vector $\mathbf{x_A} = [x_{ai}, x_{a2}, ..., x_a]^T$, we will get the extended variable vector $\mathbf{x_E} = [\mathbf{x}; \mathbf{x_A}]$. And the augmented variables can be calculated by solving the following equation:

$$\mathbf{B_{eq}x_E} = \mathbf{0} \tag{5.15}$$

where $\mathbf{B_{eq}}$ is constructed by Equation (5.14). An example of the $\mathbf{B_{eq}}$ for Figure 5.2 is shown as following:

$$\mathbf{B_{eq}} = \begin{array}{c|ccccccccccccccccc} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} & x_{12} & x_{a1} & x_{a2} & x_{a3} & x_{a6} & x_a \\ \hline node(1) & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ node(2) & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ node(3) & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ node(4) & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ sum & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & -1 \end{array} \tag{5.16}$$

Now the balance function in Equation (5.13) can be rewritten as:

$$\begin{bmatrix} \mathbf{A_{eq}} & \mathbf{0} \\ \mathbf{B_{eq}} & \end{bmatrix} \mathbf{x_E} = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \tag{5.17}$$

where we denote the extended $\mathbf{A_{E_{eq}}}$ and $\mathbf{b_{E_{eq}}}$ as

$$\mathbf{A_{E_{eq}}} = \begin{bmatrix} \mathbf{A_{eq}} & \mathbf{0} \\ \mathbf{B_{eq}} & \end{bmatrix}, \mathbf{b_{E_{eq}}} = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \tag{5.18}$$

and we also extend $\mathbf{D}$ and $\mathbf{C}$ to $\mathbf{D_E}$ and $\mathbf{C_E}$ as

$$\mathbf{D_E} = \begin{bmatrix} \mathbf{D} & \mathbf{0} \end{bmatrix}, \mathbf{C_E} = \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} \tag{5.19}$$

The final missing piece now is to get the number of transition constraint into play, which is simply:

$$x_a \leq TR \tag{5.20}$$

in matrix representation as:

$$\mathbf{A_{E_{neq}}} \mathbf{x_E} \leq TR \tag{5.21}$$

in this example we have

$$\mathbf{A_{E_{neq}}} = \begin{bmatrix} 0 & 0 & ... & 1 \end{bmatrix} \tag{5.22}$$

Now the network flow problem in Equation (5.7) can be rewritten in matrix format as:

$$\textit{Minimize } \mathbf{D_E} \mathbf{x_E} \tag{5.23a}$$

subject to

$$\mathbf{C_E} \mathbf{x_E} \leq B \tag{5.23b}$$

$$\mathbf{A_{E_{eq}}} \mathbf{x_E} = \mathbf{b_{E_{eq}}} \tag{5.23c}$$

$$\mathbf{A_{E_{neq}}} \mathbf{x_E} \leq TR \tag{5.23d}$$

$$\mathbf{X_E} \text{ is a binary vector} \tag{5.23e}$$

All this can be easily translated into CPLEX as in Listing 5.4:

```
cplex = Cplex('netflow');
cplex.Model.sense = 'minimize';
cplex.Param.lpmethod.Cur = 3; % Network simplex
cplex.addCols(D_E',[],[],[],char(ones([1 length(D_E)])*('B')));
cplex.addRows(-inf,C_E,B);
cplex.addRows(b_E_eq,A_E_eq,b_E_eq);
cplex.addRows(-inf,A_E_neq,TR)
```

Listing 5.4: CPLEX code example for network flow example

Example Result

The example network in Figure 5.2 solved by CPLEX will generate the optimal solution as:

$$\mathbf{x} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{5.24}$$

which tells us the optimal solution will choose the path x_1, x_3, x_8,and x_{12}, with the total number of transition $x_a = 1$ and total capital cost 4, which is in compliance with our investigation. The resulted solution is illustrated in Figure 5.3, where the optimal path is in solid bold lines.

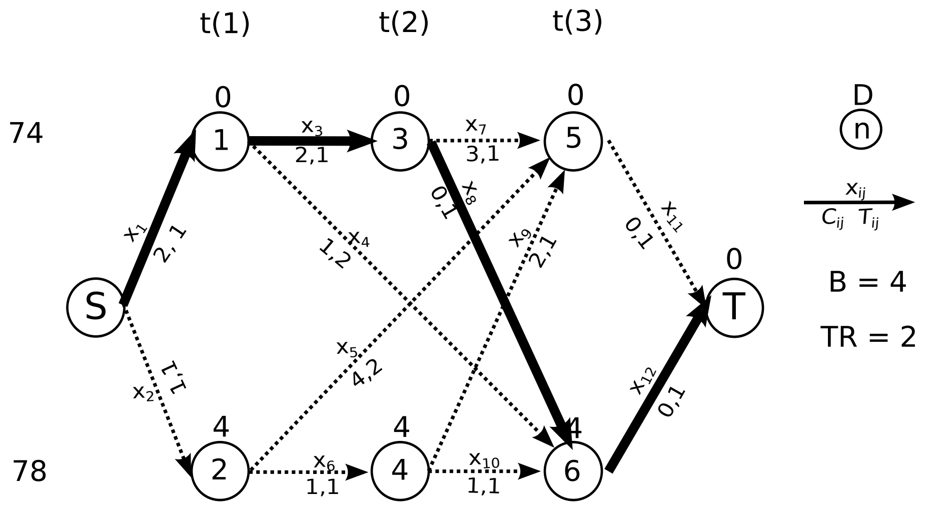


Figure 5.3: A simple network example with optimal solution

CHAPTER 6

Experiments and Results

To verify the feasibility of the proposed methods, a set of simulation experiments are carried out. In this chapter we describe the experiments setup, and discuss the results. We start by introducing the experiment setup, followed by the experiment results for all proposed components, *i.e.* indoor temperature prediction engine, HVAC load prediction engine, and temperature setpoint optimization engine. We conclude this chapter with discussion on the experiment results.

6.1 Experiment setup

6.1.1 Simulation Setup

Simulations are conducted to evaluate both the proposed temperature/load prediction algorithm and the optimization methodology. Data collected from three single-family homes and one office space in Arizona were used as shown in Table 6.1, where $SF_i, i = \{1, 2, 3\}$ indicates three single-family homes and $OF_j, j = 1$ indicates one office space respectively. Table 6.1 also shows the time span of the collected data, the date set length as well as the sampling intervals. Note that for SF_1 the sampling interval was one minute before December 2012 and five minute thereafter.

As stated previously, the data fields include indoor temperature, outdoor temperature, HVAC cooling/heating set point temperature and HVAC cooling/heating usage. And the data are divided into epochs where each epoch holds one month length of sampled data, each data epoch is then split into a training set and a test set at a 70:30 ratio for validation purposes.

Moreover, for optimization validation, the cost estimation engine takes into ac-

Data Source	Start[a]	End	Length (Months)	Interval (Minutes)
SF_1	07/2011	12/2012	18	1
SF_1	01/2013	11/2013	11	5
SF_2	01/2013	12/2013	12	5
SF_3	03/2013	10/2013	6	5
OF_1	04/2013	08/2013	5	5

Table 6.1: Collected data source

[a] Start and End dates in MM/YYYY

	Summer (May-Oct)	Winter (Nov-Apr)
Base Power Supply Charge(per KWh)	\$0.033198	\$0.025698
Deliver Charge (per KWh)		
First 500KWh	\$0.046925	\$0.047369
Next 3000KWh	\$0.068960	\$0.067309
3501KWh and above	\$0.088960	\$0.087309

Table 6.2: Residential pricing plan R-01 based on season and energy consumption

count the time-of-use pricing policies at the local utility company (Tucson Electric Power (TEP), 2012b). Monthly cost is associated with the total amount of energy utilized (kilowatt-hour), in addition to the time at which energy is utilized (on-peak, shoulder-peak, off-peak). An example is shown in Table 6.2.

6.1.2 Evaluation Metrics

For the proposed prediction engine, Mean Square Error (MSE) is utilized as an evaluation metric to indicate the average of the square of errors

$$MSE = E[(\tilde{w} - w)^2] \tag{6.1}$$

In addition, since this work aims at performing HVAC usage prediction to estimate the cumulative energy cost of HVAC operation, we introduce another metric Relative Sum of Errors (RSE), defined as:

$$RSE = \frac{\sum_t \tilde{w}(t) - \sum_t w(t)}{\sum_t w(t)} \tag{6.2}$$

which represents the cumulative errors of prediction relative to the total usage.

Moreover, to reflect the “low load” situation, we define average load of HVAC operation as

$$f_{load} = \frac{\text{Total Time HVAC is ON}}{\text{Total Time}} \tag{6.3}$$

where a bigger f_{load} means a higher load demand, and vice versa.

To evaluate the proposed optimization methodology, two evaluation metrics are utilized – the averaged set point temperature difference (ATD) and the reduced predicted cost (RPC). The ATD metric is simply the mean difference between the optimized set point schedule and the user specified set point schedule over the to-be-optimized time scope, as shown in Equation (6.4). The ATD metric will evaluate the quality of the optimized set point schedule in meeting the desired comfort level. Note that any deviation from the specified schedule results in a penalty, even if the optimization sets the cooling set point below the user specified value.

$$ATD = \frac{\sum_{j=1}^{N} |Set_j - Desired_j|}{N} \tag{6.4}$$

The RPC metric is defined as,

$$RPC = \frac{\sum_j OptCost_j - \sum_j OrigCost_j}{\sum_j OrigCost_j} \times 100\% \tag{6.5}$$

where $OrigCost$ is corresponds to the cost of a non-optimized set point schedule over the given time horizon (10 days) and $OptCost$ corresponds to the cost the optimized set point schedule generated by the optimization component. A negative RPC value indicates a percent of savings over the non-optimized set point schedule.

Because the main goal of this proposed work is to 1) estimate the HVAC load and 2) optimize the setpoint temperature, the performance of indoor temperature prediction is not evaluated individually. Rather, the outcome of the indoor temperature engine is evaluated with the HVAC load prediction, as it acts as the predecessor of the HVAC load prediction.

For both the prediction engine and optimization engine, a computational time metric T_{comp} is utilized to gauge the computational demand for both of the algorithms.

6.2 Experiment results

We first present the HVAC load prediction without indoor temperature feature (denoted as load prediction), as stated in Section 4.2, and we highlight its deficiency when the “low load problem” occurs. Then we show the revised load prediction engine with indoor temperature prediction correction (denoted as revised load prediction), described in Section 4.4, as well as its feasibility to address the “low load problem”. Finally the result of optimization engine is presented and discussed.

6.2.1 Load Prediction

The proposed load prediction algorithm was performed on the test data set with a 70:30 training-test split as stated previously. More specifically, the training data was first loaded by the prediction engine to calculate the linear regression of both the equilibrium sections and ramp section, then these calculation results were deployed on each data points in the test data set to predict the HVAC usage. Note that in reality at each specific sample the HVAC usage may be either 100% for HVAC ON or 0% for HVAC OFF, which is due to the intrinsic working status of HVAC system, and our prediction result provides an estimation in the sense of “mostly likely AVERAGE HVAC usage”. Figure 6.1 shows the simulation result on the data sampled from building SF_1 in October,2011 with the obtained MSE and RSE are 17.67 and -7.01%, respectively.

Table 6.3 shows the simulation result of the sampled data for July to September 2011 with MSE and RSE calculated for each month.

Although the results in Table 6.3 provide a satisfying RSE for the summer sea-

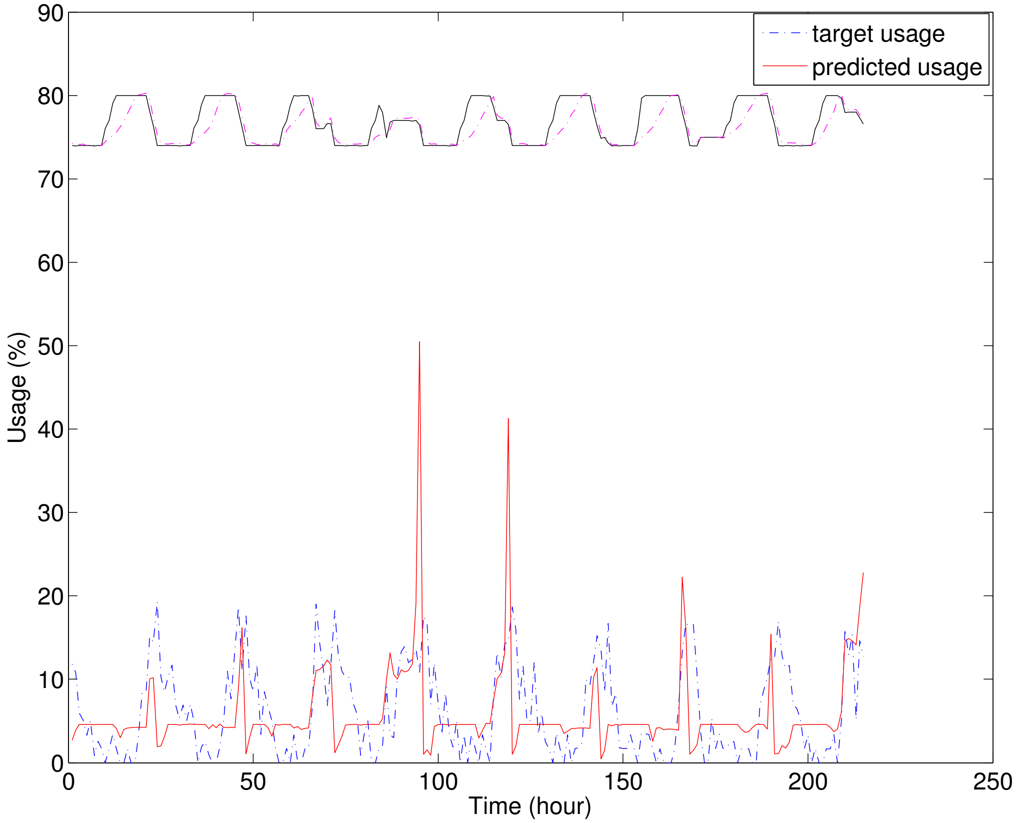


Figure 6.1: Simulation on data sampled from building SF_1 in August,2011

son in Arizona when cooling demand is high, this mechanism is challenged when the average load of HVAC operation is low. Table 6.4 shows the result of HVAC heating load prediction for the same target building in winter season, with large prediction errors. Therefore the following revised load prediction mechanism with indoor temperature estimation is utilized.

6.2.2 Load Prediction Revised

The revised load prediction engine comprises an additional indoor temperature prediction engine that provides an estimation of future indoor temperature for the HVAC load prediction to address the “low load problem”.

Month	July	August	September
MSE	30.36	28.46	29.29
RSE	7.22%	5.81%	12.48%

Table 6.3: Prediction simulation result: without indoor temperature prediction (Summer)

Month	November	December	January	February
RSE	201%	797%	178%	378%

Table 6.4: Prediction simulation result: without indoor temperature prediction (Winter)

Indoor temperature prediction

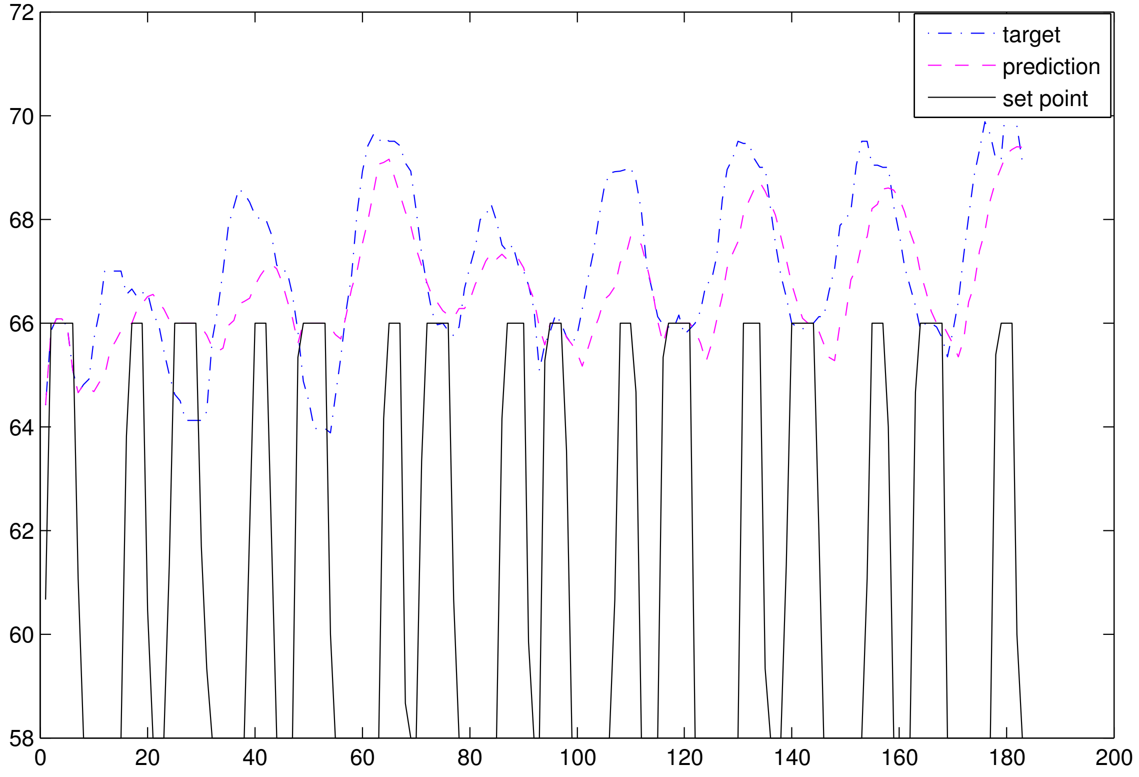


Figure 6.2: Indoor Prediction

Figure 6.2 shows an example output of the ARMAX model based indoor temperature estimation, with a model fit of 74%. The dotted-dash line represents the actual indoor temperature in the time window, while the dash line represents the predicted indoor temperature, based on the model described in Section 4.4.1.

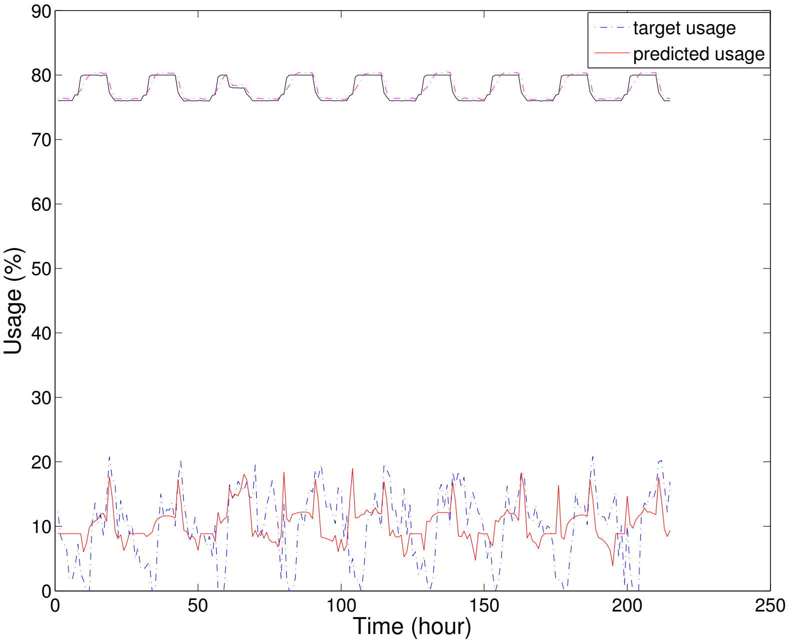
One of the artifacts to be noted is that the predicted indoor temperature is "clapped" to the setpoint temperature if it violated the setpoint, such artifacts are depicted in Figure 6.2, as around time 98 and 120. This "clapping" is forced by the indoor temperature prediction engine to emulate the effects of HVAC operation, thus provides a reasonable estimation.

Another artifact is that sometimes the actual indoor temperature itself would violate the user setpoint, as in the same Figure 6.2 where $time = 50$ and $time = 28$, the actual indoor temperature drops below the user specified heating point. This behavior might be due to an HVAC system failure and/or gas burner failure. For the current system setup and framework we did not address this problem, however, it will be fairly easy to incorporate a system failure detection component that captures such behavior and compensates for the prediction results.

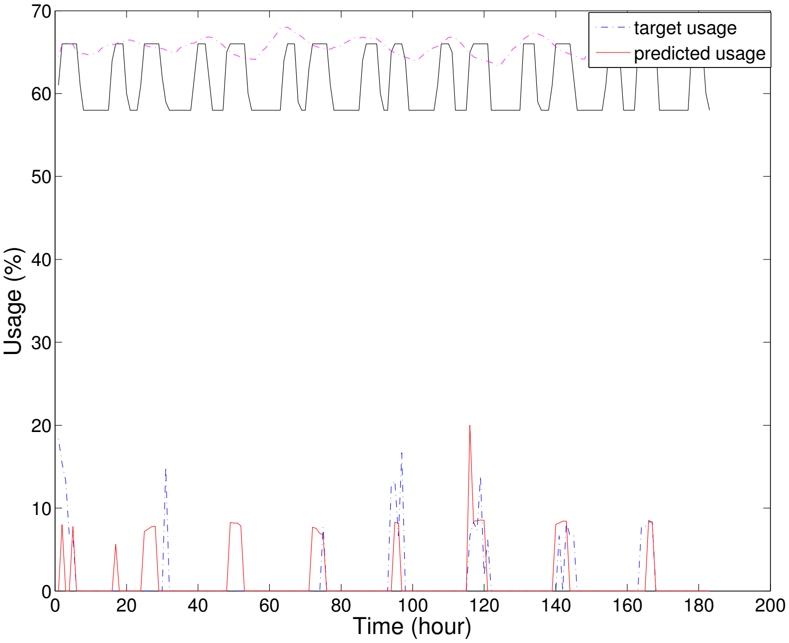
HVAC load prediction with indoor temperature estimation

With the estimated indoor temperature, HVAC load prediction is then carried out with the ability to handle the "low load problem". Figure 6.3 shows an example of HVAC load prediction result for a (roughly) 10-day horizon, where Figure 6.3a is the output for the last 10 days of September, 2013 of building SF_1, with an RSE of 3.18%, meaning the predicted load is 3.18% more than the actual load, and Figure 6.3b the last 8 days of February, 2013 of building SF_2, with an RSE of -1.45%, meaning the predicted load is 1.45% less than the actual load. Additional results can be found in Appendix A.

Moreover, the proposed scheme is applied to the collected data described in Table 6.1. The resulting MSE, RSE, load factor and work mode are recorded. Table 6.5 shows the results for the building SF_1, more results for other buildings



(a)



(b)

Figure 6.3: (a) Load prediction of last 10 days of September, 2013, SF_1 (b) Load prediction of last 8 days of February, 2013, SF_2

(SF_2, SF_3, OF_1) can be found in Appendix B.[1] [2]

Figure 6.4 shows the distribution of RSE values for building SF_1. From the result we can observe that most of the RSE is bounded in $[-15\%, 15\%]$, with mean $\mu_{RSE} = -1.41$ and standard deviation $\sigma_{RSE} = 8.33$. Result for the other test buildings can be found in Appendix C and summerized in Table 6.6.

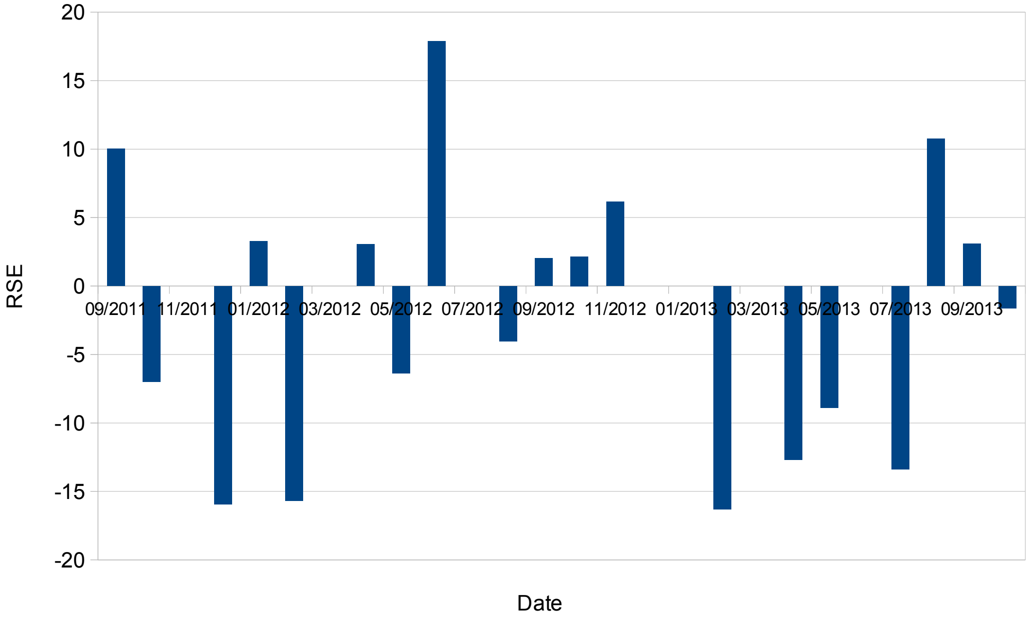


Figure 6.4: Load prediction result of SF_1

6.2.3 Optimization

Based on the results from prediction engine, the optimization engine is able to generate a network flow model and produce an optimized setpoint schedule to provide to users. Simulations show the effectiveness of optimizing the setpoints.

Figure 6.5a illustrates the original and optimized set points over a 7 day period. The original schedule reflects the non-optimized schedule of time-varying set points

[1] Due to sensor failures and/or network failures, some samples are missing or invalid, as indicated in the result

[2] March and November are the typical time of year when HVAC operation is nearly zero at the testing site, therefore the corresponding HVAC usage prediction is not performed

SF_1				
Date	MSE	RSE (%)	f_{load} (%)	Mode
07/2011	24.8673	4.8265	11.3820	C
08/2011	21.0601	-1.076	10.7517	C
09/2011	20.4265	10.0339	10.703	C
10/2011	17.6692	-6.9748	6.2425	C
11/2011	-	-	1.7624	C
12/2011	-	-	0.2447	H
01/2012	5.2366	3.2669	1.635	H
02/2012	10.079	-15.6843	1.6994	H
03/2012	-	-	0.5511	H
04/2012	15.6678	3.055	2.6924	C
05/2012	18.3329	-6.3613	5.2456	C
06/2012	17.686	17.8812	5.2747	C
07/2012[a]	na	na	na	C
08/2012	25.611	-4.0512	12.641	C
09/2012	20.2492	2.0172	10.216	C
10/2012	17.8369	2.1607	7.5061	C
11/2012	10.6454	6.1453	2.3978	C
12/2012[b]	na	na	na	H
01/2013[c]	na	na	na	H
02/2013	16.3647	-16.2934	12.9513	H
03/2013	-	-	0.7123	H
04/2013	14.1961	-12.7021	3.2612	C
05/2013	25.4567	-8.9419	18.4879	C
06/2013[d]	na	na	na	C
07/2013	21.3592	-13.3726	30.2769	C
08/2013	32.5242	10.7593	27.7473	C
09/2013	22.2903	3.1776	22.3296	C
10/2013	13.0819	-1.581	6.0432	C
11/2013	-	-	0.9475	C

Table 6.5: Prediction simulation result: with indoor temperature prediction (SF_1)

[a] invalid data due to sensor failure

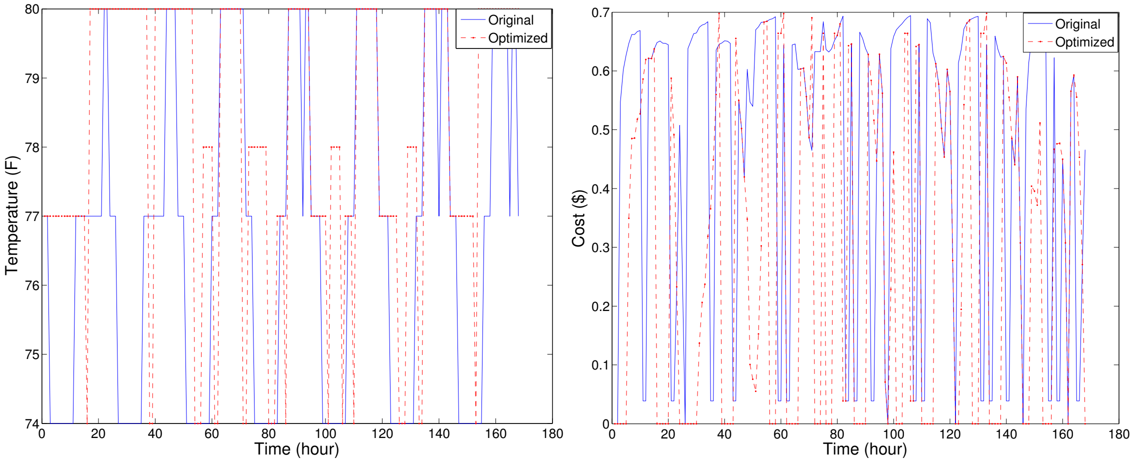
[b] missing data

[c] missing data

[d] invalid data due to sensor failure

Building	RSE Bound	μ_{RSE}	σ_{RSE}
SF_1	$[-16.3\%, 17.9\%]$	1.41	8.33
SF_2	$[-6.0\%, 8.0\%]$	0.26	3.93
SF_3	$[-10.5\%, 8.0\%]$	-0.39	6.12
OF_1	$[-10.0\%, 0\%]$	-5.32	3.06

Table 6.6: RSE results of target buildings



(a) (b)

Figure 6.5: Simulation results comparing (a) the original and optimized set point temperatures given a budget constraint of \$50 and number of transition of 28 for a 7 day time horizon, and (b) the original and optimized cost per hour of the set point schedules, total cost for both are \$78 and \$49

defined by the end user (i.e. no cost constraint). The optimized set points are the resulting set point schedule given a cost constraint of \$50 (\$200/month). In an effort to meet the user defined cost constraint the optimized set point schedule deviates slightly from the user-specified set point schedule, yielding an ATD value of 1.49°F.

Figure 6.5b compares the cost associated with the original and optimized set point schedule. As the cost constraint is the dominant factor, adherence to the user defined specified set point is sacrificed to ensure the cost constraint is met. In the example above $ATD > 0$, indicating a change in the user defined set point schedule. By making these changes we meet the \$50 cost constraint, but also achieve a RPC value of - 38%. The negative value indicates a cost saving of 38% over the estimated cost of the original set point schedule.

The engine optimizes the setpoint schedule in three ways: 1) relax the setpoint value by setting it to a lower-cost setpoint (set to a higher value in case of cooling and a lower value in case of heating, 2) relax the setpoint time by extending the time of a lower-cost setpoint, or 3) take a proactive approach, such as to "pre-cool" the house to avoid higher grid price and/or take advantage of the ramping sections where the HVAC usage is nearly zero and the indoor temperature would naturally flow with the outside temperature.

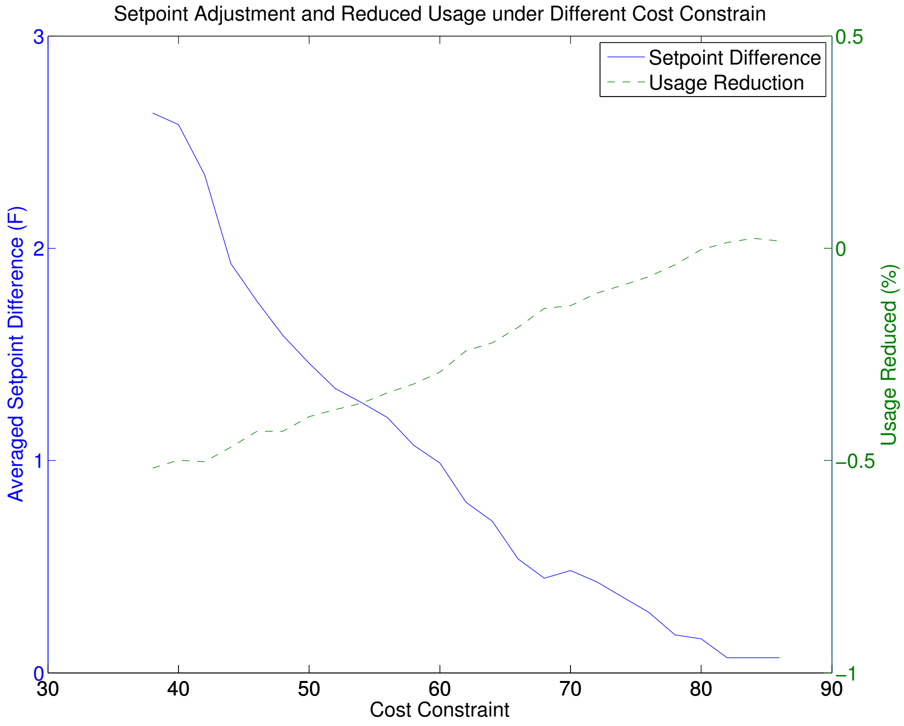


Figure 6.6: The ATD and RPC given varying cost constraints.

Figure 6.6 alternatively considers the trade-off between the ATD and RPC metrics as the cost constraint is varied from \$38 to \$86, again for a 7 day time horizon. It is not surprising to see that as the cost constraint is increased, the ATD metric decreases as the optimization component has more flexibility in finding an optimized set point schedule which adheres to the user defined values. Given the original user defined set point schedule, we can see that a cost constraint of \$82 is sufficient to achieve the desired values, and any budget increase beyond this has no benefit. Note that in Figure 6.6 the ATD and RPC value is not converged to zero, as artificial disturbance is employed within the prediction component.

Conversely, a cost constraint below \$38 will yield a non-feasible solution, and the optimization engine would output a warning message to notify the user to either adjust the budget or set point schedule. Because the network flow problem defined limited (m) possible set point values within the design space, and infeasible solution indicates that utilizing the maximum set point values for all times within the time horizon considered will not achieve the user specified cost constraint. We note however, that if we increase the number candidate set points, a solution can always be found.

6.3 Discussion

6.3.1 Accuracy

As shown in the experimentation, the proposed approach has a bounded RSE for each target building, where the bound for building SF_1 is approximately in $[-15\%, 15\%]$, and for other buildings this bound is in the range of $[-10\%, 10\%]$. Moreover, the average RSE for the target buildings are 1.41%, 0.26%, -0.39% and -5.32% respectively, meaning an average accuracy of 94% to 99%, with a nominal prediction horizon of 10 days. Moreover, this work could potentially be extended for a much larger prediction horizon, based on valid outdoor temperature information in that horizon.

6.3.2 Adaptivity

The proposed framework is also be able to address the “low load” problem, where the HVAC load factor is taken into account and handled by an estimation of indoor temperature. By doing this the proposed methods are more versatile for variant target buildings in different locations and/or weather conditions.

Another feature associated with adaptivity of the proposed framework is the scheme built in for recalculating the system model as well as the optimization, such that the system model is updated based on the most recent data, to give the user a more accurate prediction and optimization result.

6.3.3 Optimality

The linear nature of the network flow setup ensures the optimality of the optimization engine, given that the network problem setup has at least one feasible solution. (Ahuja et al., 2003)

6.3.4 Scalability

The proposed methods can be scaled to other systems if these systems hold similar characteristics in terms of system dynamics, equilibrium properties, and linearization properties. For example, the proposed methods can be easily applied to an energy management system in a building including HVAC unit as well as other appliances, to give the user estimations and optimization on how all the appliances are behaving in the building, or, in a route planning example, to give a real-time optimal route based on the estimation of traffic information.

6.3.5 Computational Demand

The computational demand is composed of the prediction and optimization processes. The prediction process has polynomial complexity and requires bounded time and memory space. The same is true for the minimum cost flow problem as we carefully select the algorithm to solve this optimization problem (Bewley, 1989; Orlin, 1985). For the simulations described above, the average processing time required to obtain an optimized 7 day set point schedule is 0.04 seconds, on an 2.4GHz Intel Dual-Core processor. We additionally implemented the optimization algorithm to execute on a mobile platform with an Intel Celeron 750 MHz processor, where an processing time of 2 seconds is required. In the current application, set points are evaluated once per minute, therefore the computational time needed for optimization is insignificant compared to the frequency at which the algorithm runs.

CHAPTER 7

Closure

7.1 Conclusion

Interacting with complex systems and making decisions to guide those systems are a growing class of problems. To interact with these systems it's of benefits for users to have some feedback information available to ensure a well-formed and sometimes more reasonable decision are made. For example, controlling the gas pedal in an automotive through an instant fuel efficiency feedback (MPG information displayed on the dashboard) of the automotive. Another example is determining an HVAC set point temperature schedule as described earlier.

To provide those feedbacks usually means a system model would be needed. On the other hand, although linear systems preserve nice properties and characteristics to be analyzed and studied, in the real world, most systems are nonlinear, and have complex system dynamics that are very difficult to model, if they can be modeled at all.

Another challenge faced by building up a model for a complex systems is that it's usually difficult to have a general system model to describe arbitrary target system. Often times each individual target system has specific parameters/behaviors that are hard to obtain in the modeling phase. For example, the parameters to describe a building would highly depend on the structure, material, layout, location of the target building, which makes it impossible to have a generic model for all buildings.

To address those problems, we discussed a data-driven approach that approximates a complex system and provides feedback to users as a guidance for them to make any decisions. This approach has the following features:

- A Data-driven approach is employed to provide an estimation of the target system output. By the data-driven nature, this work adapts to each individual target system regardless of their system-specific configurations and parameters, instead, all the parameters will be learned by the system through the machine learning process, which makes this approach generative.

- The target system is modeled from data sampled by existing and usually inexpensive sensors, therefore reduce the needs for additional infrastructural investment on sensors and/or actuators.

- A *mid-term* prediction/estimation horizon is provided in this work, as for the HVAC usage application typical prediction horizon of ten days is employed, which provides a holistic view of the system behavior in a larger scope than most of the existing one-step ahead solutions.

- Feedback information on the system behavior is provided by this work to the users to make well-informed decisions. Moreover, an integrated optimization engine is also developed to calculate an optimal/sub-optimal setpoints as an supplemental suggestion that helps users make decisions. Furthermore, this optimization as well as prediction process can be performed “on-line”, therefore give users an “up-to-date” information on the system operation cost as well as decision options.

- The approximation and regression approach in this work is computational inexpensive and suitable for implementation on embedded devices, therefore suitable for integrating into existing infrastructure as well as emergent technologies.

We discussed the formation of this approach mathematically and illustrated its application in details for an HVAC load prediction/optimization framework.

In the discussed HVAC management framework, an Averaged Sectional Model (ASM) is developed to model the system through sampled HVAC usage and environmental data. A minimum cost network flow formulation is utilized to generate the optimal set point temperatures under a user-specified cost constraint.

Simulations on data sampled from three single-family homes and one office space in Arizona area are conducted. It is demonstrated that this framework can effectively provide an optimized set point temperature and forecasted cost, with averaged Relative Sum of Errors (RSE) 1.41%, 0.26%, -0.39% and -5.32% on each target building respectively. We also demonstrate the optimization engine that provides set point temperature schedule suggestions based on the HVAC usage estimation, and a user cost constraint. In addition, the computation and memory requirements for this algorithm are suitable for embedded system implementation, where a 2 seconds computation time is required on an Intel Celeron 750MHz single-core processor, which is insignificant to the sampling time typically at five minutes. With this framework, a meaningful interpretation of HVAC usage is provided to users, as well as the ability to visualize the long term energy and cost consequences to make well informed decisions.

7.2 Future Work

In this work we demonstrate the feasibility of the proposed method on data sampled from four buildings in Arizona area. It would be beneficial to gather additional data from other locations with different parameters such as weather pattern, usage profile and building structure, and verify the performance of the proposed methods.

Another future work is to apply the method at higher tiers for building energy systems. For example, this method can be applied to estimate and optimize the overall home energy consumption including HVAC operation and other appliances, providing feedback information on the energy cost and/or appliances performance, and suggestions on the schedule of these appliances to reduce energy consumption; or at the community tier that provides relative information and suggestions on energy consumption in a neighborhood, or even at the regional or national tier. Moreover, with the emergent development of smart grid/smart home technology, Distributed Energy Generation (DEG) units such as solar panels and small wind turbines are becoming pervasive. These units often have a dynamic energy profile that can be affected by factors from installation location to temperature to manufactoring differences, which make them hard to be modeled using a dedicated model. With the help of the proposed system approximation and optimization scheme, however, these system can be easily factored in and approximated properly within the overall system, providing useful feedback information for users.

Furthermore, this method can be utilized for demand response analysis. Demand Response (DR) is defined as: (Balijepalli et al., 2011) "Changes in electric usage by end-use customers from their normal consumption patterns in response to changes in the price of electricity over time, or to incentive payments designed to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized." The proposed method running on different tiers could provide insights on the energy usage profiles of end-use customers, and help utility providers make decisions to manage energy generation and reduce cost.

APPENDIX A

Prediction Examples

The following figures illustrate part of the example outputs of HVAC usage prediction for the target buildings SF_1, SF_2, SF_3 and OF_1 in this work.

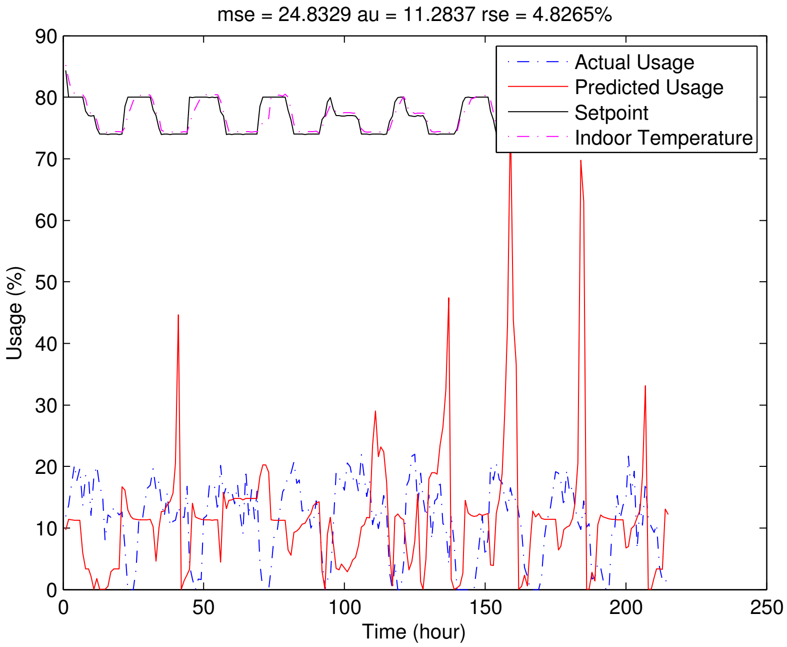


Figure A.1: Load prediction of last 10 days of July, 2011, SF_1

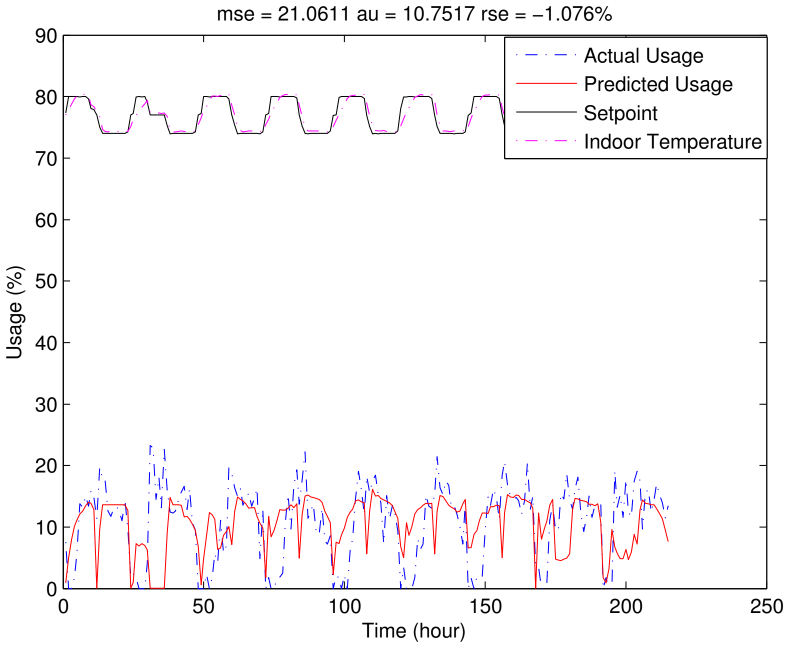


Figure A.2: Load prediction of last 10 days of August, 2011, SF_1

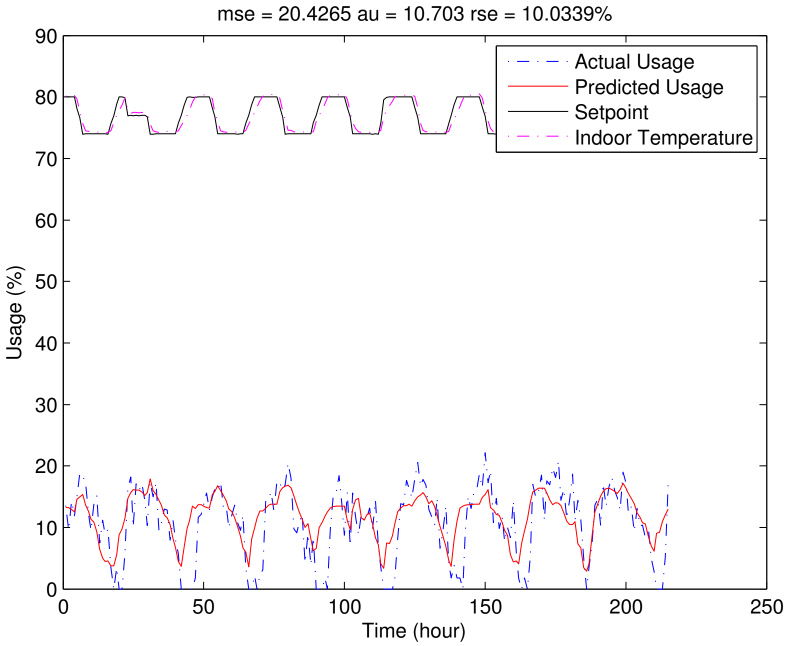


Figure A.3: Load prediction of last 10 days of September, 2011, SF_1

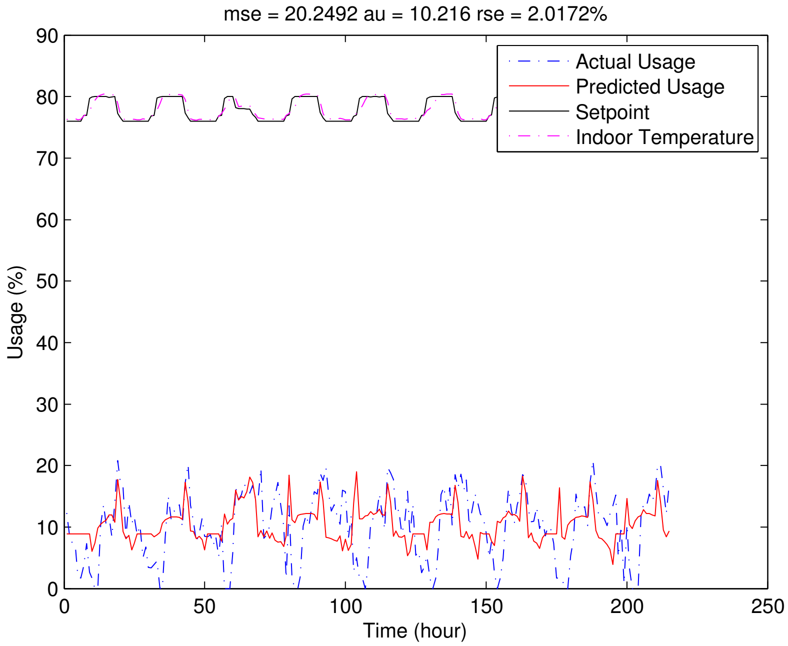


Figure A.4: Load prediction of last 10 days of September, 2012, SF_1

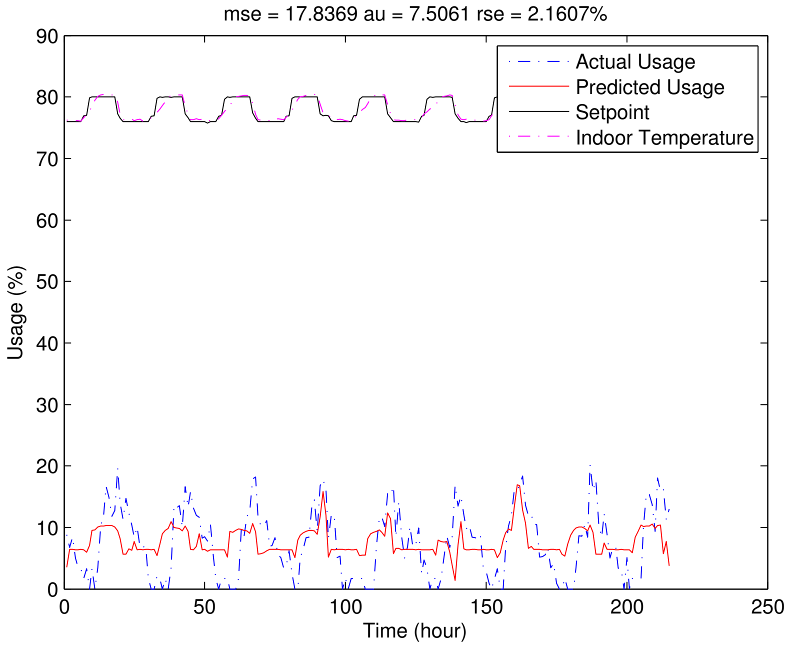


Figure A.5: Load prediction of last 10 days of October, 2012, SF_1

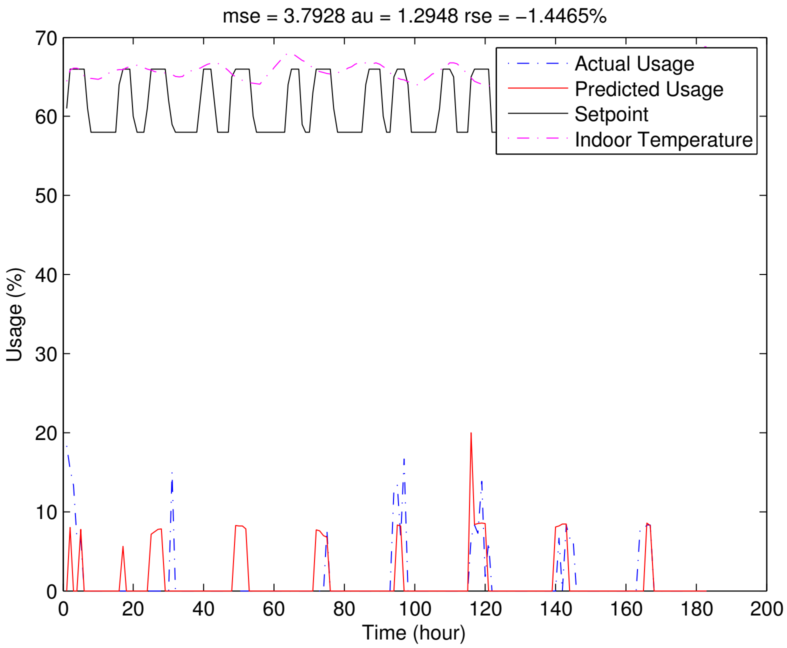


Figure A.6: Load prediction of last 10 days of Februray, 2013, SF_2

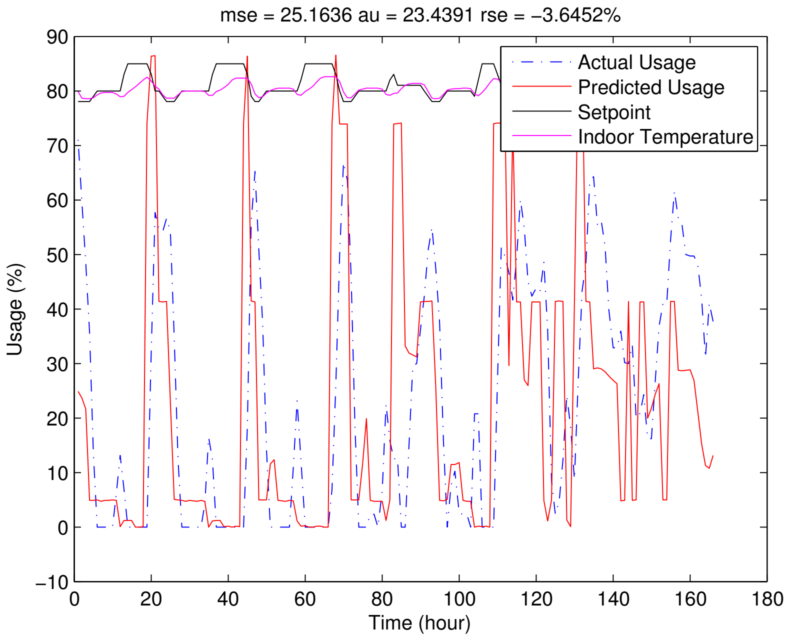


Figure A.7: Load prediction of last 10 days of June, 2013, SF_2

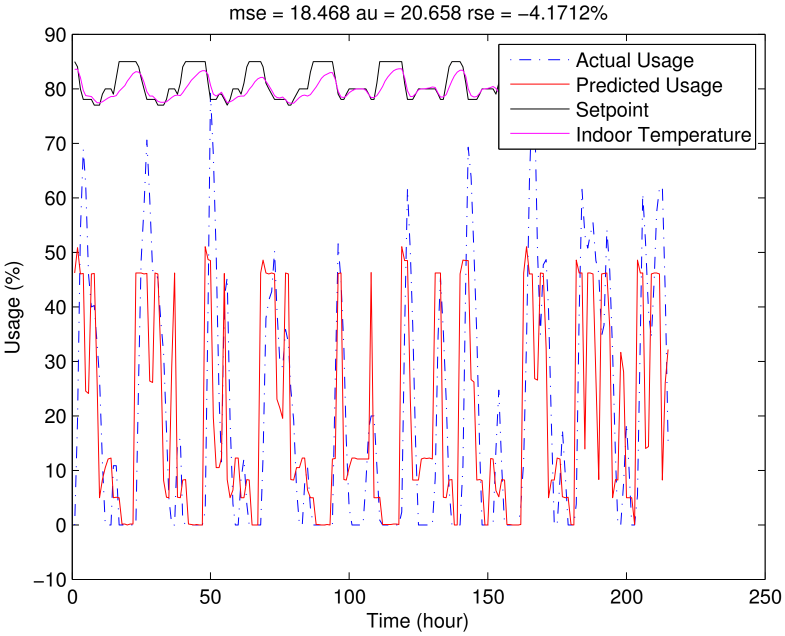


Figure A.8: Load prediction of last 10 days of August, 2013, SF_2

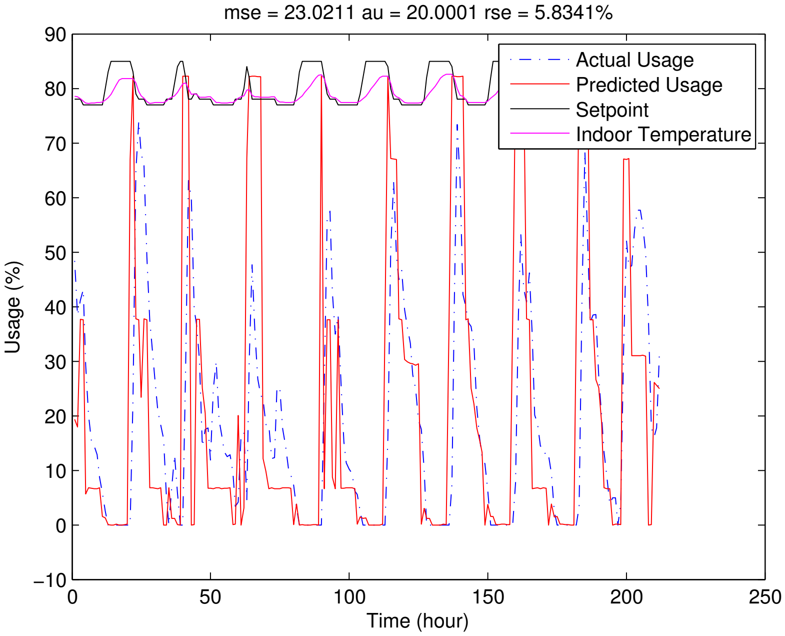


Figure A.9: Load prediction of last 10 days of September, 2013, SF_2

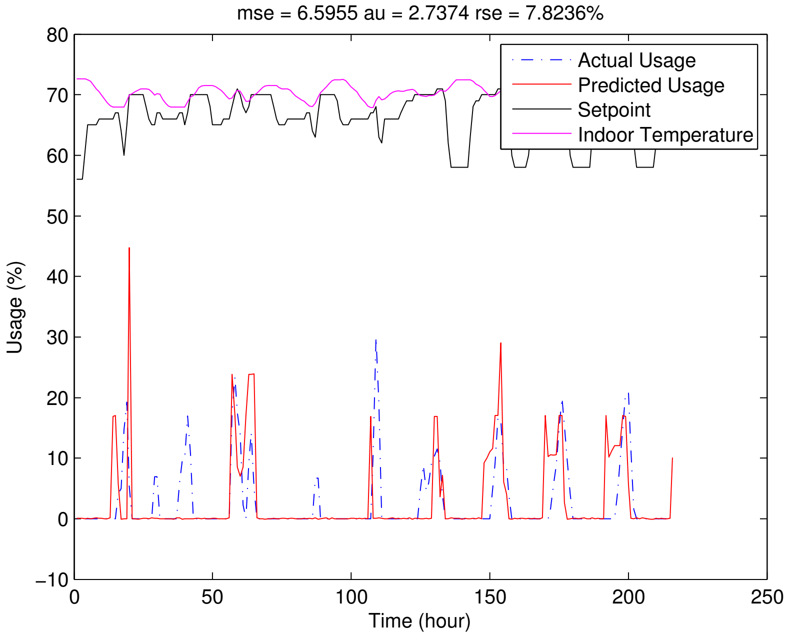


Figure A.10: Load prediction of last 10 days of December, 2013, SF_2

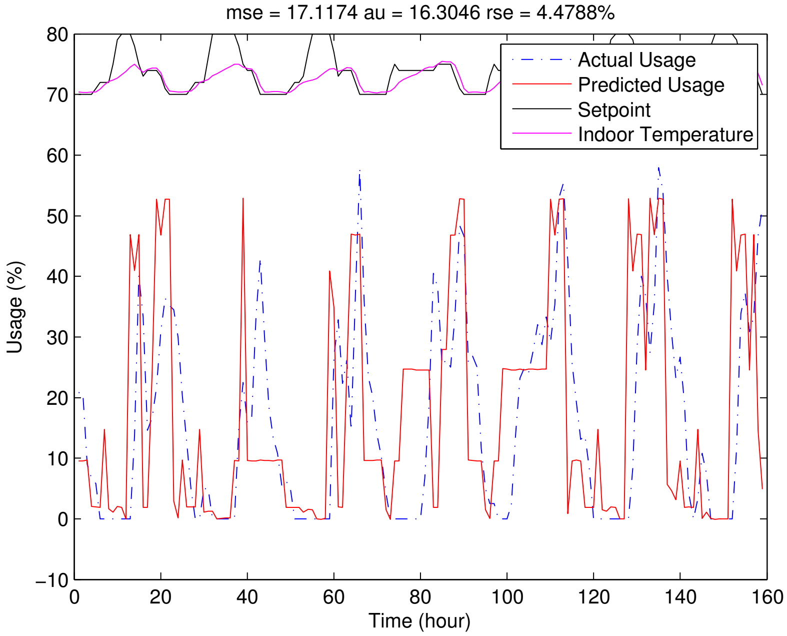


Figure A.11: Load prediction of last 10 days of April, 2013, SF_3

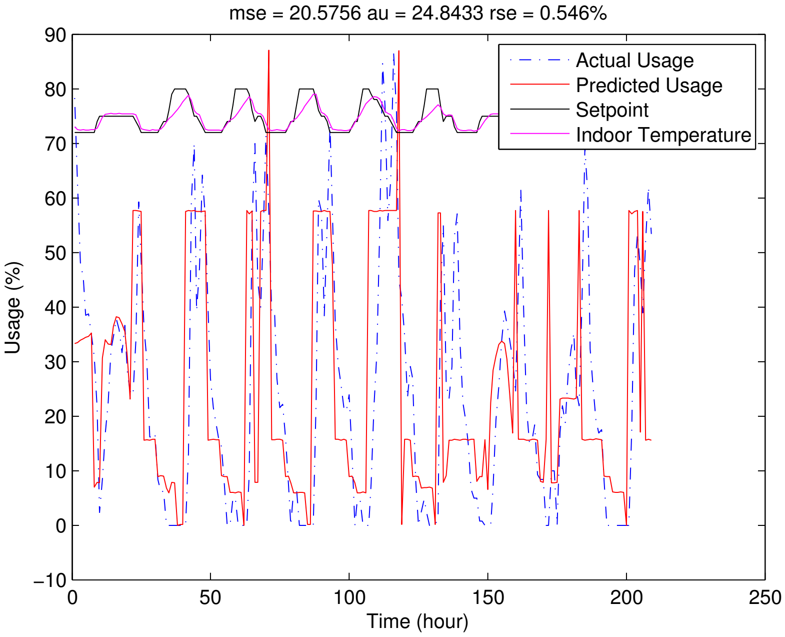


Figure A.12: Load prediction of last 10 days of September, 2013, SF_3

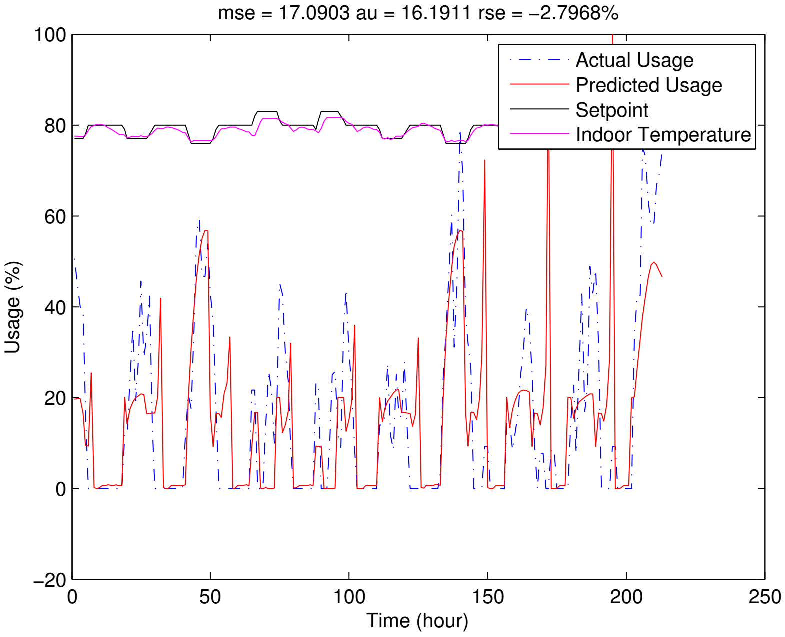


Figure A.13: Load prediction of last 10 days of May, 2013, OF_1

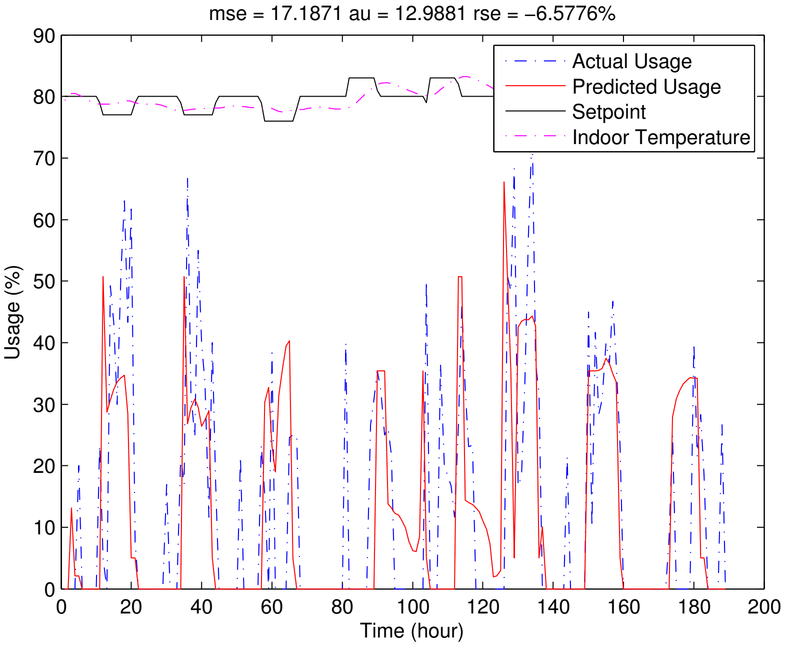


Figure A.14: Load prediction of last 10 days of June, 2013, OF_1

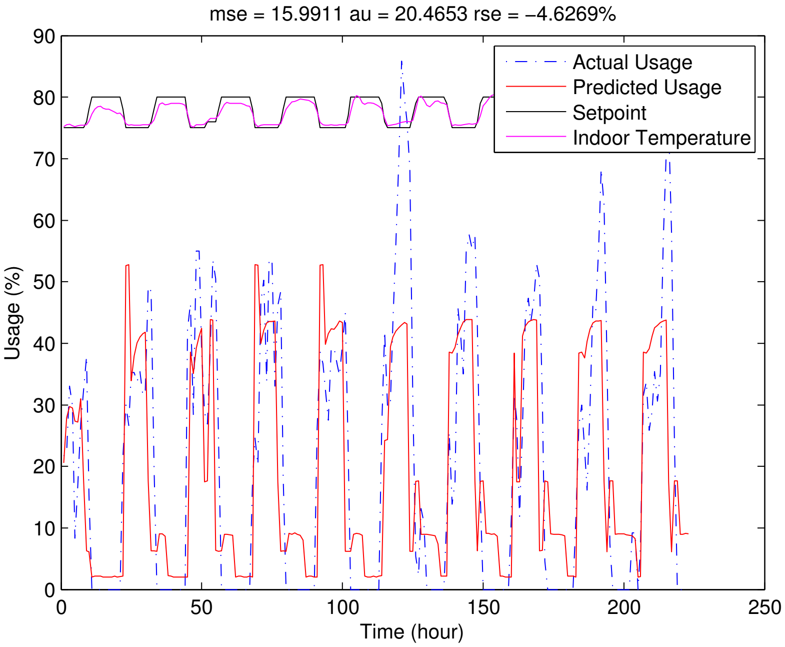


Figure A.15: Load prediction of last 10 days of July, 2013, OF_1

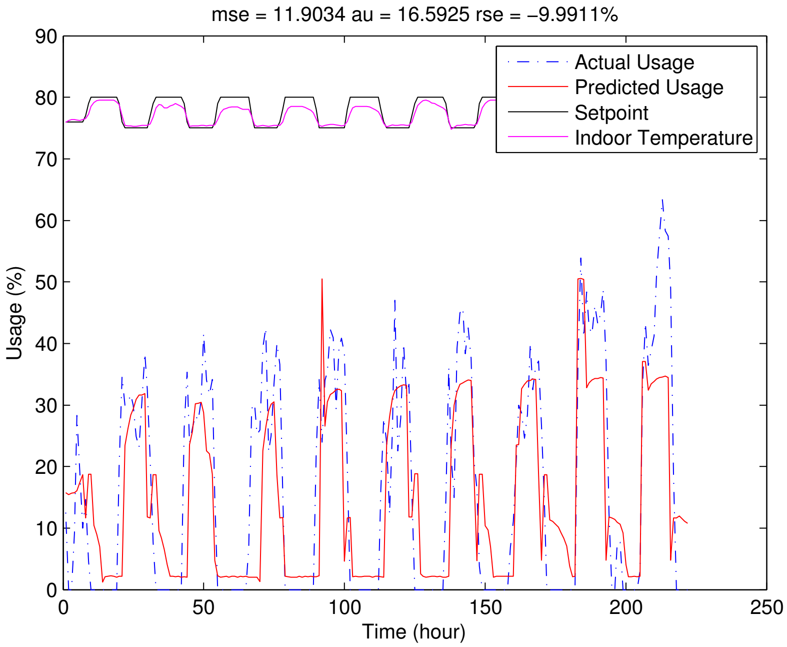


Figure A.16: Load prediction of last 10 days of August, 2013, OF_1

APPENDIX B

Prediction Results

The following tables show the HVAC prediction result for the target buildings in this work. Table B.1 shows results for an office space OF_1, while Table B.2, Table B.3 and Table B.4 show results for three single family houses SF_1, SF_2 and SF_3 respectively.

OF_1				
Date	MSE	RSE (%)	f_{load} (%)	Mode
04/2013	9.2283	-2.6088	6.1047	C
05/2013	17.09	-2.7968	16.1911	C
06/2013	17.1871	-6.5776	12.9881	C
07/2013	15.9911	-4.6269	20.4653	C
08/2013	11.9034	-9.9911	16.5925	C

Table B.1: Prediction simulation result: with indoor temperature prediction (OF_1)

SF_1				
Date	MSE	RSE (%)	f_{load} (%)	Mode
07/2011	24.8673	4.8265	11.3820	C
08/2011	21.0601	-1.076	10.7517	C
09/2011	20.4265	10.0339	10.703	C
10/2011	17.6692	-6.9748	6.2425	C
11/2011	-	-	1.7624	C
12/2011	-	-	0.2447	H
01/2012	5.2366	3.2669	1.635	H
02/2012	10.079	-15.6843	1.6994	H
03/2012	-	-	0.5511	H
04/2012	15.6678	3.055	2.6924	C
05/2012	18.3329	-6.3613	5.2456	C
06/2012	17.686	17.8812	5.2747	C
07/2012[a]	na	na	na	C
08/2012	25.611	-4.0512	12.641	C
09/2012	20.2492	2.0172	10.216	C
10/2012	17.8369	2.1607	7.5061	C
11/2012	10.6454	6.1453	2.3978	C
12/2012[b]	na	na	na	H
01/2013[c]	na	na	na	H
02/2013	16.3647	-16.2934	12.9513	H
03/2013	-	-	0.7123	H
04/2013	14.1961	-12.7021	3.2612	C
05/2013	25.4567	-8.9419	18.4879	C
06/2013[d]	na	na	na	C
07/2013	21.3592	-13.3726	30.2769	C
08/2013	32.5242	10.7593	27.7473	C
09/2013	22.2903	3.1776	22.3296	C
10/2013	13.0819	-1.581	6.0432	C
11/2013	-	-	0.9475	C

Table B.2: Prediction simulation result: with indoor temperature prediction (SF_1)

[a] invalid data due to sensor failure

[b] missing data

[c] missing data

[d] invalid data due to sensor failure

SF_2				
Date	MSE	RSE (%)	f_{load} (%)	Mode
01/2013	-	-	0.9129	H
02/2013	3.7928	-1.4465	1.2948	H
03/2013	-	-	0.3297	H
04/2013[a]	na	na	na	C
05/2013	26.5236	-5.4374	20.0865	C
06/2013	25.1636	-3.6452	23.4391	C
07/2013	24.7576	2.7812	27.0244	C
08/2013	18.468	-4.1712	20.658	C
09/2013	23.0211	5.8341	20.0001	C
10/2013	17.8531	2.6283	13.2033	C
11/2013	-	-	0.4968	C
12/2013	6.5955	7.8236	2.7374	H

Table B.3: Prediction simulation result: with indoor temperature prediction (SF_2)

[a] invalid data due to sensor failure

SF_3				
Date	MSE	RSE (%)	f_{load} (%)	Mode
03/2013	18.2799	-10.2204	8.5539	C
04/2013	17.1174	4.4788	16.3046	C
05/2013[a]	na	na	na	C
06/2013	15.6407	-8.5831	8.8169	C
07/2013	35.1689	7.4628	39.6892	C
08/2013[b]	na	na	na	C
09/2013	20.5756	0.546	24.8433	C
10/2013	14.1496	3.1626	12.3644	C

Table B.4: Prediction simulation result: with indoor temperature prediction (SF_3)

[a] invalid data due to sensor failure
[b] invalid data due to sensor failure

APPENDIX C

RSE Charts

The following figures show the HVAC prediction RSE result for the target buildings in this work.

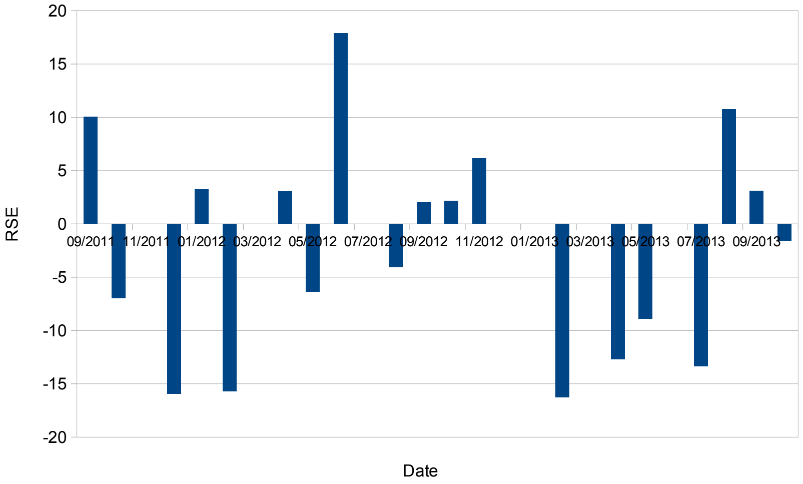


Figure C.1: Load prediction result of SF_1

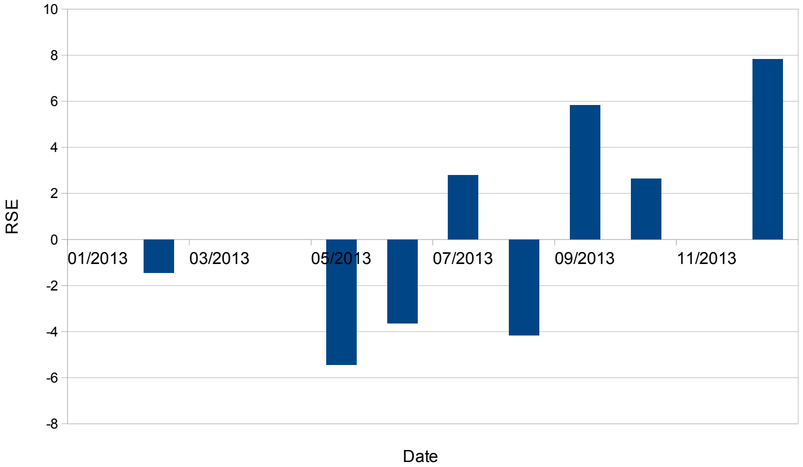


Figure C.2: Load prediction result of SF_2

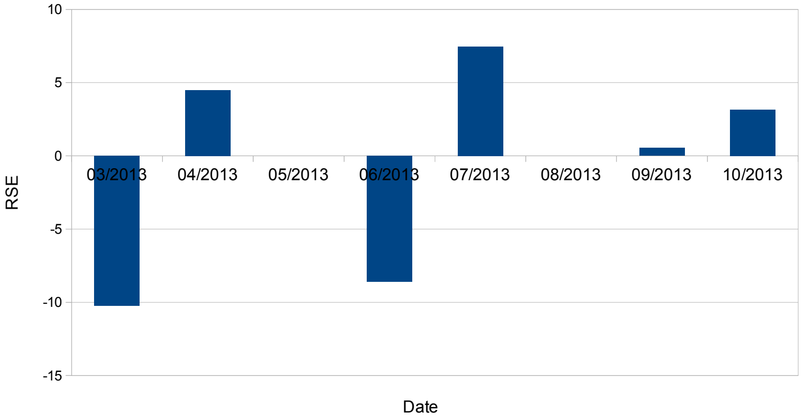


Figure C.3: Load prediction result of SF_3

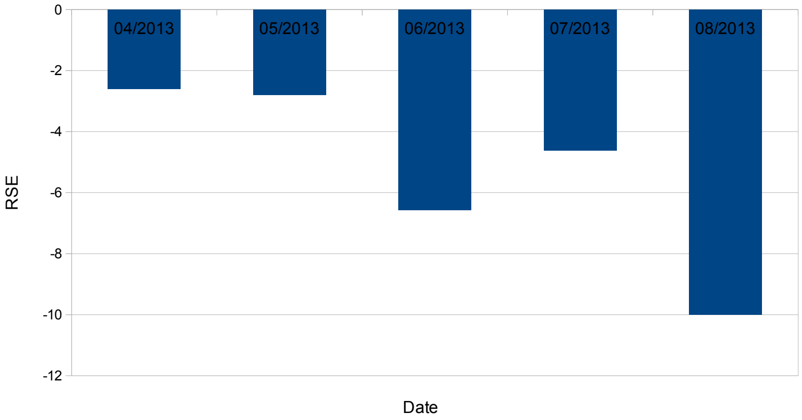


Figure C.4: Load prediction result of OF_1

REFERENCES

Abdi, H. and L. J. Williams (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, **2**(4), pp. 433–459.

Ahuja, R., T. Magnanti, and J. Orlin (2003). *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall.

Allgöwer, F. and A. Zhen (2000). *Nonlinear model predictive control*, volume 26. Springer.

Alternative Software Concept (2012). AEPS. Http://www.alteps.com.

Amjady, N. (2001). Short-term hourly load forecasting using time-series modeling with peak load estimation capability. *Power Systems, IEEE Transactions on*, **16**(4), pp. 798 –805. ISSN 0885-8950. doi:10.1109/59.962429.

Architectural Energy Corp. (2012). REM/Design. Http://www.archenergy.com/products/remdesign.

ASHRAE (1996). HVAC systems and equipment. *American Society of Heating, Refrigerating, and Air Conditioning Engineers, Atlanta, GA*.

ASHRAE (2001). Energy Estimating and Modeling Methods. *ASHRAE Fundamentals*.

Aswani, A., N. Master, J. Taneja, A. Krioukov, D. Culler, and C. Tomlin (2012a). Energy-Efficient Building HVAC Control Using Hybrid System LBMPC. *arXiv preprint arXiv:1204.4717*.

Aswani, A., N. Master, J. Taneja, V. Smith, A. Krioukov, D. Culler, and C. Tomlin (2012b). Identifying models of HVAC systems using semiparametric regression. In *American Control Conference (ACC), 2012*, pp. 3675–3680. ISSN 0743-1619.

Athans, M. (1971). The role and use of the stochastic linear-quadratic-Gaussian problem in control system design. *Automatic Control, IEEE Transactions on*, **16**(6), pp. 529–552.

Autodesk Inc. (2010). Green Building Studio. Http://usa.autodesk.com/green-building-studio.

Avci, M., M. Erkoc, A. Rahmani, and S. Asfour (2013). Model predictive HVAC load control in buildings using real-time electricity pricing. *Energy and Buildings*, **60**, pp. 199–209. ISSN 03787788. doi:10.1016/j.enbuild.2013.01.008.

Balijepalli, V. M., V. Pradhan, S. Khaparde, and R. Shereef (2011). Review of demand response under smart grid paradigm. In *Innovative Smart Grid Technologies-India (ISGT India), 2011 IEEE PES*, pp. 236–243. IEEE.

Beghi, A., L. Cecchinato, M. Rampazzo, and F. Simmini (2010). Load forecasting for the efficient energy management of HVAC systems. *Sustainable Energy Technologies (ICSET), 2010 IEEE International Conference on*, pp. 1–6. doi: 10.1109/ICSET.2010.5684414.

Ben-Nakhi, A. E. and M. A. Mahmoud (2004). Cooling load prediction for buildings using general regression neural networks. *Energy Conversion and Management*, **45**, pp. 2127–2141. ISSN 0196-8904. doi:10.1016/j.enconman.2003.10.009.

Bewley, T. F. (1989). *Advances in Economic Theory*. Cambridge University Press.

Camacho, E. F. and C. B. Alba (2013). *Model predictive control*. Springer.

Camacho, E. F. and C. Bordons (2004). *Model predictive control*, volume 2. Springer London.

Catalina, T., V. Iordache, and B. Caracaleanu (2013). Multiple regression model for fast prediction of the heating energy demand. *Energy and Buildings*, **57**, pp. 302–312. ISSN 03787788. doi:10.1016/j.enbuild.2012.11.010.

Chen, Y., P. B. Luh, C. Guan, Y. Zhao, L. D. Michel, M. A. Coolbeth, P. B. Friedland, S. J. Rourke, and S. Member (2010). Short-Term Load Forecasting : Similar Day-Based Wavelet Neural Networks. *IEEE Transactions on Power Systems*, **25**(1), pp. 322–330.

Cherkassky, V., S. R. Chowdhury, V. Landenberger, S. Tewari, and P. Bursch (2011). Prediction of electric power consumption for commercial buildings. *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pp. 666–672. ISSN 2161-4393. doi:10.1109/IJCNN.2011.6033285.

Chirarattananon, S. and J. Taveekun (2004). An OTTV-based energy estimation model for commercial buildings in Thailand. *Energy and Buildings*, **36**(7), pp. 680–689. ISSN 0378-7788. doi:10.1016/j.enbuild.2004.01.035.

Chiu, Y.-C., S. Gao, D.-Y. Lin, and X. Hu (2011). A Routing Behavior Model for Vacant Taxi Cabs in Urban Traffic Networks. In *The 11th Asia-Pacific ITS Forum & Exhibition*.

Clarke, J., J. Cockroft, S. Conner, J. Hand, N. Kelly, R. Moore, T. O'Brien, and P. Strachan (2002). Simulation-assisted control in building energy management systems. *Energy and buildings*, **34**(9), pp. 933–940.

Cui, P., H. Yang, J. D. Spitler, and Z. Fang (2008). Simulation of hybrid ground-coupled heat pump with domestic hot water heating systems using HVACSIM+. *Energy and Buildings*, **40**(9), pp. 1731–1736.

Daou, K., R. Wang, and Z. Xia (2006). Desiccant cooling air conditioning: a review. *Renewable and Sustainable Energy Reviews*, **10**(2), pp. 55 – 77. ISSN 1364-0321. doi:http://dx.doi.org/10.1016/j.rser.2004.09.010.

Darghouth, N. R., G. Barbose, and R. Wiser (2011). The impact of rate design and net metering on the bill savings from distributed PV for residential customers in California. *Energy Policy*, **39**(9), pp. 5243–5253. ISSN 0301-4215. doi:http://dx.doi.org/10.1016/j.enpol.2011.05.040.

Department of Energy (2005). Energy Policy Act of 2005.

Department of Energy (2012). Energy Efficiency and Renewable Energy. Http://apps1.eere.energy.gov/buildings/energyplus.

Dong, B., C. Cao, and S. E. Lee (2005). Applying support vector machines to predict building energy consumption in tropical region. *Energy and Buildings*, **37**(5), pp. 545–553.

Dorf, R. C. (1995). *Modern control systems*. Addison-Wesley Longman Publishing Co., Inc.

Electricity Advisory Committee (EAC) (December 2008). SmartGrid: Enabler of the New Energy Economy. *A Report by The Electricity Advisory Committee*.

Escrivá-Escrivá, G., C. Álvarez Bel, C. Roldán-Blay, and M. Alcázar-Ortega (2011). New artificial neural network prediction method for electrical consumption forecasting based on building end-uses. *Energy and Buildings*, **43**(11), pp. 3112–3119. ISSN 03787788. doi:10.1016/j.enbuild.2011.08.008.

Filtrete (2013). Remote Access Thermostat.

Fischer, C. (2008). Feedback on household electricity consumption: a tool for saving energy? *Energy Efficiency*, **1**(1), pp. 79–104. ISSN 1570-646X. doi:10.1007/s12053-008-9009-7.

Fong, K., V. Hanby, and T. Chow (2006). HVAC system optimization for energy management by evolutionary programming. *Energy and Buildings*, **38**(3), pp. 220–231. ISSN 03787788. doi:10.1016/j.enbuild.2005.05.008.

Fong, K., V. Hanby, and T. Chow (2009). System optimization for HVAC energy management using the robust evolutionary algorithm. *Applied Thermal Engineering*, **29**(11-12), pp. 2327 – 2334. ISSN 1359-4311. doi:10.1016/j.applthermaleng.2008.11.019.

Froehlich, J. (2009). Promoting Energy Efficient Behaviors in the Home through Feedback: The Role of Human Computer Interaction. *HCIC 2009 Winter Workshop.*

FRONTLINE (2010). California Crisis Timeline.

Gao, G. and K. Whitehouse (2009). The self-programming thermostat: optimizing setback schedules based on home occupancy patterns. *ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys)*, pp. 67–72.

Garcia, C. E., D. M. Prett, and M. Morari (1989). Model predictive control: theory and practice—a survey. *Automatica*, **25**(3), pp. 335–348.

Golub, G. H. and C. Reinsch (1970). Singular value decomposition and least squares solutions. *Numerische Mathematik*, **14**(5), pp. 403–420.

González, P. a. and J. M. Zamarreño (2005). Prediction of hourly energy consumption in buildings based on a feedback artificial neural network. *Energy and Buildings*, **37**(6), pp. 595–601. ISSN 03787788. doi:10.1016/j.enbuild.2004.09.006.

Gonzalez, P. A. and J. M. Zamarreno (2005). Prediction of hourly energy consumption in buildings based on a feedback artificial neural network. *Energy and Buildings*, **37**(6), pp. 595–601. ISSN 0378-7788. doi:10.1016/j.enbuild.2004.09.006.

Guan, C., P. B. Luh, M. A. Coolbeth, Y. Zhao, L. D. Michel, Y. Chen, S. Member, C. J. Manville, P. B. Friedland, and S. J. Rourke (2009). Very Short-term Load Forecasting : Multilevel Wavelet Neural Networks with Data Pre-filtering. pp. 1–8.

Gugliermetti, F., G. Passerini, and F. Bisegna (2004). Climate models for the assessment of office buildings energy performance. *Building and Environment*, **39**(1), pp. 39–50. ISSN 0360-1323. doi:10.1016/S0360-1323(03)00138-0.

Hargreaves, T., M. Nye, and J. Burgess (2010). Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors. *Energy Policy*, **38**(10), pp. 6111–6119. ISSN 0301-4215. doi:10.1016/j.enpol.2010.05.068.

He, X.-D., S. Liu, and H. H. Asada (1997). Modeling of vapor compression cycles for multivariable feedback control of HVAC systems. *Journal of dynamic systems, measurement, and control*, **119**(2), pp. 183–191.

Howell, R. H., W. J. Coad, and H. J. Sauer (2009). *Principles of Heating, Ventilating, and Air Conditioning: A Textbook with Design Data Based on the 2009 Ashrae Handbook: Fundamentals.* Atlanta, GA: American Society of Heating, Refrigerating and Air-Conditioning Engineers.

Hu, X. (2013). Developing the Analysis Methodology and Platform for Behaviorally Induced System Optimal Traffic Management.

Jiang, X., P. Dutta, D. Culler, and I. Stoica (2007). Micro Power Meter for Energy Monitoring of Wireless Sensor Networks at Scale. In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, IPSN '07, pp. 186–195. ACM, New York, NY, USA. ISBN 978-1-59593-638-7. doi: 10.1145/1236360.1236386.

Jin, G.-Y., W.-J. Cai, L. Lu, E. L. Lee, and A. Chiang (2007). A simplified modeling of mechanical cooling tower for control and optimization of HVAC systems. *Energy conversion and management*, **48**(2), pp. 355–365.

Jolliffe, I. (2005). *Principal component analysis*. Wiley Online Library.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, **82**(1), pp. 35–45.

Karatasou, S., M. Santamouris, and V. Geros (2006a). Modeling and predicting building's energy use with artificial neural networks: Methods and results. *Energy and Buildings*, **38**(8), pp. 949–958.

Karatasou, S., M. Santamouris, and V. Geros (2006b). Prediction of energy consumption in buildings with artificial intelligent techniques and Chaos time series analysis. *International Workshop on Energy Performance and Environmental Quality of Buildings*.

Katipamula, S., T. Reddy, and D. Claridge (1998). Multivariate regression modeling. *Journal of Solar Energy Engineering*, **120**(3), pp. 177–184.

Ke, Y.-P. and S. A. Mumma (1997). Optimized supply-air temperature (SAT) in variable-air-volume (VAV) systems. *Energy*, **22**(6), pp. 601–614.

Keis, K., E. Magnusson, H. Lindström, S.-E. Lindquist, and A. Hagfeldt (2002). A 5% efficient photoelectrochemical solar cell based on nanostructured ZnO electrodes. *Solar Energy Materials and Solar Cells*, **73**(1), pp. 51–58. ISSN 09270248. doi:10.1016/S0927-0248(01)00110-6.

Klema, V. and A. J. Laub (1980). The singular value decomposition: Its computation and some applications. *Automatic Control, IEEE Transactions on*, **25**(2), pp. 164–176.

Kulkarni, M. R. and F. Hong (2004). Energy optimal control of a residential space-conditioning system based on sensible heat transfer modeling. *Building and Environment*, **39**(1), pp. 31–38.

Leung, M., N. C. Tse, L. Lai, and T. Chow (2012). The use of occupancy space electrical power demand in building cooling load prediction. *Energy and Buildings*, **55**, pp. 151–163. ISSN 03787788. doi:10.1016/j.enbuild.2012.08.032.

Li, Q., Q. Meng, J. Cai, H. Yoshino, and A. Mochida (2009). Predicting hourly cooling load in the building: A comparison of support vector machine and different artificial neural networks. *Energy Conversion and Management*, **50**(1), pp. 90–96. ISSN 01968904. doi:10.1016/j.enconman.2008.08.033.

Li, Z. and G. Huang (2013). Re-evaluation of Building Cooling Load Prediction Models for Use in Humid Subtropical Area. *Energy and Buildings*, **62**, pp. 442–449. ISSN 03787788. doi:10.1016/j.enbuild.2013.03.035.

Lien, C.-H., Y.-W. Bai, H.-C. Chen, and C.-H. Hung (2009). Home appliance energy monitoring and controlling based on Power Line Communication. In *Consumer Electronics, 2009. ICCE '09. Digest of Technical Papers International Conference on*, pp. 1–2. doi:10.1109/ICCE.2009.5012315.

Lu, L., W. Cai, Y. C. Soh, L. Xie, and S. Li (2004). HVAC system optimization—condenser water loop. *Energy Conversion and Management*, **45**(4), pp. 613–630. ISSN 01968904. doi:10.1016/S0196-8904(03)00181-X.

Lu, L., W. Cai, L. Xie, S. Li, and Y. C. Soh (2005). HVAC system optimization—in-building section. *Energy and Buildings*, **37**(1), pp. 11–22. ISSN 03787788. doi:10.1016/j.enbuild.2003.12.007.

Ma, Y., F. Borrelli, B. Hencey, B. Coffey, S. Bengea, and P. Haves (2012). Model Predictive Control for the Operation of Building Cooling Systems. *Control Systems Technology, IEEE Transactions on*, **20**(3), pp. 796–803.

Mathews, E., D. Arndt, C. Piani, and E. van Heerden (2000). Developing cost efficient control strategies to ensure optimal energy use and sufficient indoor comfort. *Applied Energy*, **66**(2), pp. 135–159. ISSN 0306-2619. doi:10.1016/S0306-2619(99)00035-5.

McDowall, R. (2006). *Fundamentals of Hvac Systems*. Elsevier, Amsterdam.

McDowell, T. P., S. Emmerich, J. W. Thornton, and G. Walton (2003). Integration of airflow and energy simulation using CONTAM and TRNSYS. *Transactions-American Socienty of Heating Refrigerating and Air Conditioning Engineers*, **109**(2), pp. 757–770.

Morari, M. and J. H Lee (1999). Model predictive control: past, present and future. *Computers & Chemical Engineering*, **23**(4), pp. 667–682.

Mossolly, M., K. Ghali, and N. Ghaddar (2009). Optimal control strategy for a multi-zone air conditioning system using a genetic algorithm. *Energy*, **34**(1), pp. 58–66.

Mustafaraj, G., J. Chen, and G. Lowry (2010). Development of room temperature and relative humidity linear parametric models for an open office using BMS data. *Energy and Buildings*, **42**(3), pp. 348–356. ISSN 03787788. doi:10.1016/j.enbuild.2009.10.001.

Nassif, N., S. Kajl, and R. Sabourin (2004). Evolutionary algorithms for multiobjective optimization in HVAC system control strategy. *IEEE Annual Meeting of the Fuzzy Information Processing Society*, pp. 51–56.

Nassif, N., S. Kajl, and R. Sabourin (2005). Optimization of HVAC control system strategy using two-objective genetic algorithm. *HVAC&R Research*, **11**(3), pp. 459–486.

Nassif, N., S. Moujaes, and M. Zaheeruddin (2008). Self-tuning dynamic models of HVAC system components. *Energy and Buildings*, **40**(9), pp. 1709–1720. ISSN 03787788. doi:10.1016/j.enbuild.2008.02.026.

NEST Lab (2011). http://www.nest.com.

Norman L. Miller, J. J., Katharine Hayhoe and M. Auffhammer (2008). Climate, Extreme Heat, and Electricity Demand in California. *Journal of Applied Meteorology and Climatology*, **47**, p. 1834–1844.

Oldewurtel, F., A. Parisio, C. N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and M. Morari (2012). Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings*, **45**, pp. 15–27.

Orlin, J. B. (1985). A Polynomial Time Dual Simplex Algorithm for the Minimum Cost Flow Problem. *Symposium on Mathematical Programming.*

Peng, X. and a.H.C. van Paassen (1998). A state space model for predicting and controlling the temperature responses of indoor air zones. *Energy and Buildings*, **28**(2), pp. 197–203. ISSN 03787788. doi:10.1016/S0378-7788(98)00021-8.

Privara, S., Z. Vana, D. Gyalistras, J. Cigler, C. Sagerschnig, M. Morari, and L. Ferkl (2011). Modeling and identification of a large multi-zone office building. In *2011 IEEE International Conference on Control Applications (CCA)*, pp. 55–60. IEEE.

Richter, S., C. N. Jones, and M. Morari (2009). Real-time input-constrained MPC using fast gradient methods. In *Decision and Control, 2009 held jointly with the*

2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on, pp. 7387–7393.

Riederer, P., D. Marchio, J. Visier, A. Husaunndee, and R. Lahrech (2002a). Room thermal modelling adapted to the test of HVAC control systems. *Building and Environment*, **37**(8-9), pp. 777–790. ISSN 03601323. doi:10.1016/S0360-1323(02)00052-5.

Riederer, P., D. Marchio, and J. C. Visier (2002b). Influence of sensor position in building thermal control: criteria for zone models. *Energy and Buildings*, **34**(8), pp. 785–798. ISSN 03787788. doi:10.1016/S0378-7788(02)00097-X.

Široký, J., F. Oldewurtel, J. Cigler, and S. Prívara (2011). Experimental analysis of model predictive control for an energy efficient building heating system. *Applied Energy*, **88**, pp. 3079–3087.

Söderstrom, T. and P. Stoica (1989). *System identification*. Prentice Hall International, London.

Sowell, E. F. and P. Haves (2001). Efficient solution strategies for building energy system simulation. *Energy and Buildings*, **33**(4), pp. 309–317.

Speyer, J. L. (1979). Computation and transmission requirements for a decentralized linear-quadratic-Gaussian control problem. *Automatic Control, IEEE Transactions on*, **24**(2), pp. 266–269.

Stanek, M., M. Morari, and K. Frohlich (2001). Model-aided diagnosis: an inexpensive combination of model-based and case-based condition assessment. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, **31**(2).

Sturzenegger, D., D. Gyalistras, M. Morari, and R. S. Smith (2012). Semi-automated modular modeling of buildings for model predictive control. In *BuildSys '12: Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pp. 99–106. ACM Request Permissions.

Sundramoorthy, V., G. Cooper, N. Linge, and Q. Liu (2011). Domesticating Energy-Monitoring Systems: Challenges and Design Concerns. *IEEE Pervasive Computing*, **10**(1), pp. 20–27. ISSN 1536-1268. doi:10.1109/MPRV.2010.73.

Tashtoush, B., M. Molhim, and M. Al-Rousan (2005). Dynamic model of an HVAC system for control analysis. *Energy*, **30**(10), pp. 1729–1745. ISSN 03605442. doi:10.1016/j.energy.2004.10.004.

Tucson Electric Power (TEP) (2012a). PowerShift Program.

Tucson Electric Power (TEP) (2012b). TEP Power Partners.

University of Wisconsin Madison (2012). TRNSYS:A TRaNsient SYstems Simulation Program. Http://sel.me.wisc.edu/trnsys/index.html.

U.S. Department of Energy (2008). The Smart Grid: An Introduction. Http://www.oe.energy.gov/1165.htm.

US Environmental Protection Agency (EPA) (2012). US EPA ENERGY STAR program. http://www.energystar.gov.

Wang, S. and X. Xu (2006). Parameter estimation of internal thermal mass of building dynamic models using genetic algorithm. *Energy Conversion and Management*, **47**(13-14), pp. 1927–1941.

Wang, Y.-W., W.-J. Cai, Y.-C. Soh, S.-J. Li, L. Lu, and L. Xie (2004). A simplified modeling of cooling coils for control and optimization of HVAC systems. *Energy Conversion and Management*, **45**(18), pp. 2915–2930.

Wang, Z.-S., Y. Cui, K. Hara, Y. Dan-oh, C. Kasada, and A. Shinpo (2007). A High-Light-Harvesting-Efficiency Coumarin Dye for Stable Dye-Sensitized Solar Cells. *Advanced Materials*, **19**(8), pp. 1138–1141. ISSN 1521-4095. doi:10.1002/adma.200601020.

Wemhoff, A. P. (2010). Application of optimization techniques on lumped HVAC models for energy conservation. *Energy and Buildings*, **42**(12), pp. 2445–2451.

Wu, S. and J. Sun (2012a). A physics-based linear parametric model of room temperature in office buildings. *Building and Environment*, **50**, pp. 1–9. ISSN 03601323. doi:10.1016/j.buildenv.2011.10.005.

Wu, S. and J.-Q. Sun (2012b). Multi-stage regression linear parametric models of room temperature in office buildings. *Building and Environment*, **56**, pp. 69–77. ISSN 03601323. doi:10.1016/j.buildenv.2012.02.026.

Xi, X.-C., A.-N. Poo, and S.-K. Chou (2007). Support vector regression model predictive control on a HVAC plant. *Control Engineering Practice*, **15**(8), pp. 897–908.

Yang, J., H. Rivard, and R. Zmeureanu (2005). On-line building energy prediction using adaptive artificial neural networks. *Energy and Buildings*, **37**(12), pp. 1250–1259. ISSN 0378-7788. doi:10.1016/j.enbuild.2005.02.005.

Yik, F., J. Burnett, and I. Prescott (2001). Predicting Air-conditioning Energy Consumption of A Group of Buildings using Different Heat Rejection Methods. *Energy and Buildings*, **22**, pp. 151–166.

Yun, K., R. Luck, P. J. Mago, and H. Cho (2012). Building hourly thermal load prediction using an indexed ARX model. *Energy and Buildings*, **54**, pp. 225–233. ISSN 03787788. doi:10.1016/j.enbuild.2012.08.007.

Zhou, Y., J. Wu, R. Wang, S. Shiochi, and Y. Li (2008). Simulation and experimental validation of the variable-refrigerant-volume (VRV) air-conditioning system in EnergyPlus. *Energy and buildings*, **40**(6), pp. 1041–1047.