DISTRIBUTED CONTROL OF HVAC&R NETWORKS

A Dissertation

by

MATTHEW STUART ELLIOTT

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Bryan Rasmussen
Committee Members,	Charles Culp
	Dan McAdams
	Alan Palazzolo
	John Valasek
Head of Department,	Andreas Polycarpou

August 2013

Major Subject: Mechanical Engineering

ABSTRACT

Heating, ventilation, air conditioning, and refrigeration (HVAC&R) systems are a major component of worldwide energy consumption, and frequently consist of complex networks of interconnected components. The ubiquitous nature of these systems suggests that improvements in their energy efficiency characteristics can have significant impact on global energy consumption. The complexity of the systems, however, means that decentralized control schemes will not always suffice to balance competing goals of energy efficiency and occupant comfort and safety.

This dissertation proposes control solutions for three facets of this problem. The first is a cascaded control architecture for actuators, such as electronic expansion valves, that provides excellent disturbance rejection and setpoint tracking characteristics, as well as partial nonlinearity compensation without a compensation model. The second solution is a hierarchical control architecture for multiple-evaporator vapor compression systems that uses model predictive control (MPC) at both the supervisory and component levels. The controllers leverage the characteristics of MPC to balance energy efficiency with occupant comfort. Since the local controllers are decentralized, the architecture retains a degree of modularity—changing one component does not require changing all controllers.

The final contribution is a new distributed optimization algorithm that is rooted in distributed MPC and is especially motivated by HVAC&R systems. This algorithm allows local level optimizers to iterate to a centralized solution. The optimizers have no

knowledge of any plant other than the plant they are associated with, and only need to communicate with their immediate neighbors. The efficacy of the algorithm is displayed with two sets of examples. One example is simulation based, wherein a building is modeled in the EnergyPlus software suite. The other is an experimental example. In this example, the algorithm is applied to a multiple evaporator vapor compression system. In both cases the design method is discussed, and the ability of the algorithm to reduce energy consumption when properly applied is demonstrated.

ACKNOWLEDGEMENTS

I first thank my advisor, Professor Bryan Rasmussen, for many years of excellent mentoring and financial support. We started at A&M together, and though this has been a long road, it has been a pleasure to have you as my guide along the way. I also thank my committee members, Professors Alan Palazzolo, Charles Culp, John Valasek, and Dan McAdams, for serving on my committee and mentoring me.

For much good cheer and fellowship, I thank my colleagues and partners-in-crime at the Thermo-Fluids Control Library, and there have been many (colleagues, not crimes). I send a special tip-of-the-hat to Young-Joon and Abhi, with whom I started and now finally follow out the door. I also thank the many students I had the privilege of teaching throughout my years in graduate school.

I thank my parents and family, for their endless love and support. Finally, I give thanks to the Father Above, for giving me the talents I have and for the opportunity to use them.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER I

INTRODUCTION AND LITERATURE REVIEW

Rising energy costs and an increasingly competitive economic landscape create a need for inventive and practical approaches to improve energy efficiency in large-scale systems such as the power grid and intelligent building systems. Building operations account for approximately 40% of US energy usage and carbon emissions [1, 2], and 75% of peak electrical demand [3]. Advanced control strategies carry enormous potential for payoffs from increased energy efficiency and present new challenges for engineers working in this field.

Commercial building energy systems are interconnected systems, often with centralized monitoring, although the sensors are widely distributed and individual components are independently controlled. Thus, the typical building is a "system-of-systems" with dynamics that evolve on multiple time scales with many varied performance objectives. A typical building HVAC system is shown in part in Figure 1.1. Multiple "chillers" utilize a refrigerant-based vapor compression cycle to cool a secondary fluid (e.g. water), which is pumped to various devices or zones of the building, where heat exchangers in the Air Handling Units cool the tertiary fluid, air. A network of fans and ducts then carry the cooled air to the desired locations. To reject waste heat, water is used to transport the heat produced by the condensers to cooling towers, where forced air convection removes heat to the outdoor environment. The chillers, which use a vapor compression cycle, are the heart of this system; they are

themselves interconnected systems consisting of multiple heat exchangers, expansion valves, and compressors. Their dynamics are driven by fluctuating pressures of the two-phase fluid in the heat exchangers, and fluctuating refrigerant mass flow rates through the compressor and expansion valve.

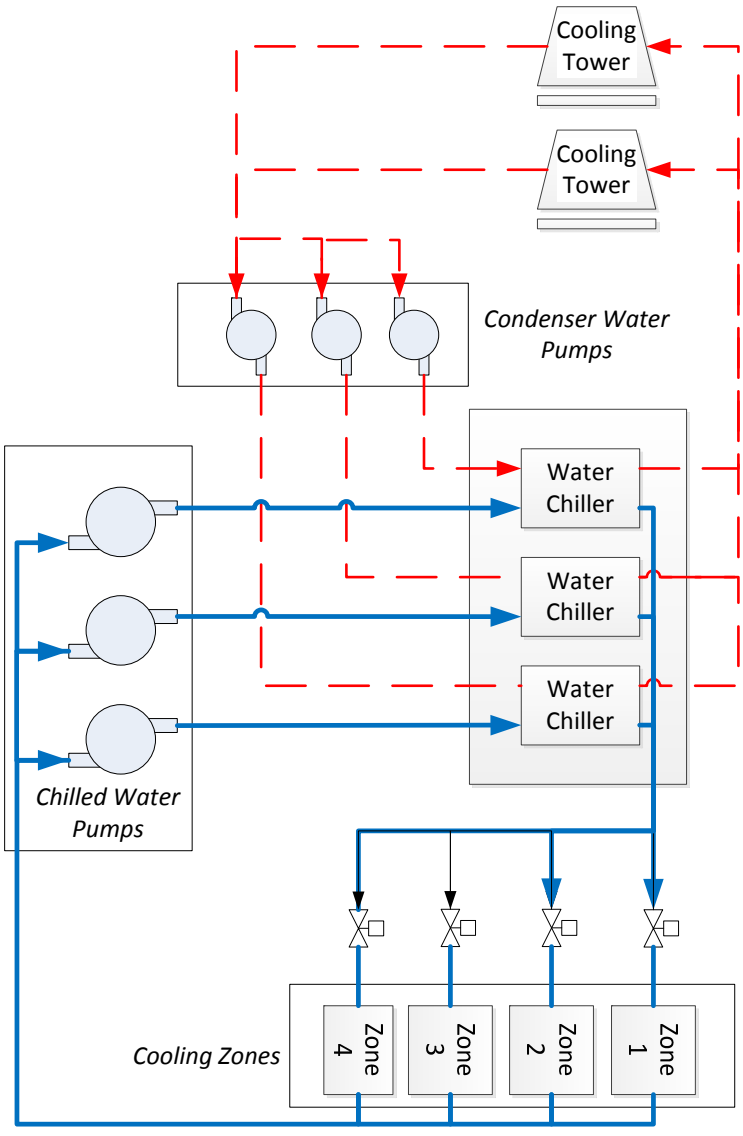


Figure 1.1: Building HVAC system

Standard practice employs separate, independent controllers for each of these varied components. Completely decentralized controllers, particularly proportional-integral (PI) controllers, are widely used in building systems, but the lack of coordination can result in inefficient operation or controllers working against each other, leading to "hunting" behavior. Centralized control, however, is practically impossible to implement for large building systems with dozens of widely distributed air handling units or evaporators. The communication requirements are very high for a centralized approach, with the possibility of controller failure if the lines of communication fail. Additionally, if a component is changed in the system the entire controller must be redesigned. An alternative to a centralized controller handling all of the component-level actuators is a hierarchical control architecture, which uses a supervisory controller to calculate setpoints for lower level controllers. This can reduce the computational load, since the global calculations can generally be made at a lower frequency than what the actuators require. This arrangement still has high communication requirements, however, since each controller must retain contact with a central controller. A preferable control approach would be a distributed architecture, where each controller is independent of the others but still retains some knowledge of the system as a whole, which gives an advantage over completely decentralized controllers.

An additional consideration of control of building systems is the constantly changing nature of the building dynamics. Components fail, and are replaced with different models. Weather creates daily and seasonal changes in operating conditions. Buildings age and windows leak. People come and go; windows and doors are opened

and closed. Rooms are reconfigured. Occupants change thermostat settings, and building operators override system controls. Maintaining a single highly detailed and accurate model of all system behaviors, or updating a centralized controller every time a component is replaced or reconfigured is infeasible; this is another reason that decentralized, independent control strategies are standard practice. Likewise fully communicative distributed control approaches where every subsystem requires perfect knowledge of all other subsystems' dynamics would place unrealistic demands on the communication and computation infrastructure. Determining a modular, flexible control approach that still achieves the optimality of a centralized control solution is critical to achieving a practical control solution for modern building systems.

This dissertation provides contributions to the solution of these problems at multiple levels. A novel approach to component control is presented in Chapter III. A hierarchical control structure that benefits from use of the model predictive control architecture is given in Chapter IV. The development of a completely distributed optimization is presented in Chapter V, and this approach is demonstrated through simulation and experiment in Chapters VI and VII, respectively.

The rest of this chapter is organized as follows. First, an overview of the extant literature is provided with respect to model predictive control (MPC) and the various decentralized and distributed forms of this control architecture. In particular, the cooperative forms of distributed MPC are discussed, since they provide the starting point for much of the work presented herein. The application of MPC to building systems and refrigeration cycles is then discussed, and followed with a section on multiple evaporator

vapor compression systems. Finally, a primer is given on the basics of the vapor compression cycle and definition of many of the concepts referred to throughout the dissertation.

Literature Review

Model Predictive Control

The control architecture used throughout this research is model predictive control (MPC). MPC is an overarching term for a suite of control strategies mostly developed in industry during the 1970s. A controller based upon one of these strategies selects control inputs via the online optimization of a predefined cost function at discrete time intervals. MPC allows for explicit handling of input, output, and state constraints—if a set of control inputs violate a constraint as predicted by the dynamic model, that set of inputs is discarded as a possible choice. MPC is also well suited for multiple-input, multiple-output (MIMO) control. Therefore, closely-coupled dynamics in the plant can be controlled with a single controller, whereas a group of single-input, single-output (SISO) controllers will frequently interfere with each other. Furthermore, since the cost function can be changed online to match changes in global or external conditions, MPC-based controllers can have a high level of flexibility in meeting general operational goals as the operating conditions change.

MPC is also referred to as receding horizon control, since the horizon for which prediction is performed moves ahead in time at each sampling instant. There are two important horizons in MPC, both of which are expressed in terms of sampling instants. The prediction horizon is the span of time for which the plant outputs are predicted. The

control horizon is the number of control inputs that are calculated in the prediction computation, and is always smaller than the prediction horizon. The size of the prediction horizon is generally limited by computation speed; it is important to choose the control horizon such that the difference between the control and prediction horizons is as least as long as it takes for all dynamics in the system to settle out [4]. The primary weakness of most advanced control techniques with respect to VCC systems is the same as that of classical control: inability to account for constraints explicitly in the controller design, leading to a lower level of performance [5]. MPC, on the other hand, can explicitly handle constraints, and has been used to solve many control problems in the chemical and refining industries [6]. Indeed, these industries specifically developed MPC to operate complex multivariable systems near constraints without violating them, since the most economic operating conditions in chemical processing are typically at intersections of constraints [7]. Academic research followed after successful industrial implementation; many of the MPC controllers in wide use are proprietary algorithms [7]. A very frequently used form of MPC is the Generalized Predictive Control (GPC) algorithm proposed by Clarke; this algorithm is perhaps the most widely used in HVAC applications [8]. Advancements in the study of stability in the presence of constraints and robustness have been made; Mayne provides a useful survey of the work performed in [9]. Some work has been done in the field of Robust MPC; the most common approach involves the solution of multiple Linear Matrix Inequalities (LMIs), as proposed by Kothare [10]. In addition, widely referenced textbooks have been published by Camacho [5] and Rossiter [4], and Clarke published a collection of MPC papers

describing advances in robustness, stability, and applications [11]. A recent textbook by Rawlings and Mayne provides an excellent overview of the current state of the art of MPC, including robust and decentralized MPC of large scale systems [12].

MPC is a control algorithm grown out of the needs of the process industries, and much of the work that continues to advance the field is motivated by these needs. The dynamic systems in these industries (e.g., chemical plants) tend to be highly complex and nonlinear, making centralized control (as shown in Figure 1.2) unrealistic due to modeling complexities and optimization difficulties. Additionally, infeasibility of the state constraints can result in closed-loop unstable behavior [13]. This issue was studied by Scokaert and Rawlings in greater depth in [14]. An early attempt at breaking up MPC into smaller, more easily solved problems involved passing information back and forth between lower-level, decentralized controllers and a supervisor; the supervisor had knowledge of the interaction dynamics between subsystems. This iteration-based process was still suboptimal with respect to the global cost function, but was faster and simpler to implement [15]. Another approach proposed by van Antwerp and Braatz involved approximating the constraint set of the MPC optimization with an ellipsoid, which allowed off-line computation of a suboptimal solution that the on-line MPC algorithm scaled as necessary [16]. Further demonstrating the connection between constraint feasibility and successful MPC implementation, Scokaert and Rawlings showed that suboptimal MPC will stabilize a nonlinear plant so long as the constraints are feasible [17].

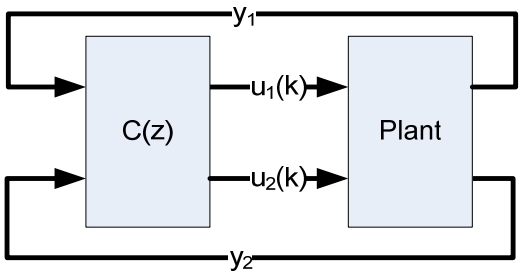


Figure 1.2: Centralized controller. The controller calculates the control signals sent to the plant at sampling instant *k*.

Decentralized MPC Architectures

Most of the preceding suboptimal MPC studies assume that the entire plant is being controlled by a single controller. Apart from the tremendous complexities of modeling an entire plant, centralized MPC is often not commercially or organizationally practical, since many of the individual units in a large dynamic network are maintained by different operators or companies [18]. Indeed, the author of [19] argues that the tendency of the controls community to seek centralized optimal controllers has proven to be unproductive and should be abandoned in favor of interconnected, decentralized approaches.

Decentralized control is generally defined as a control architecture in which the controller regulating each plant in an interconnected system has no awareness of the interconnection dynamics between plants, as shown in Figure 1.3. Interest in decentralized control has continued since the 1970s, when the increasing complexity of controlled systems sparked interest in the subject, and a need for a scientific approach to

the control design was identified [20, 21]. Design of a decentralized control architecture is a non-trivial problem, since the actions of one controller will affect the outputs of other plants. Indeed, closing the loop on one plant in the system can have significant impact on the open loop dynamics of the other subsystems, such as changing zeros from minimum to non-minimum phase [22]. Much of the current research in this field builds upon the ideas found in Siljak's monograph [23]. A recent survey of current research directions, especially those involving strongly coupled subsystems, can be found in [24].

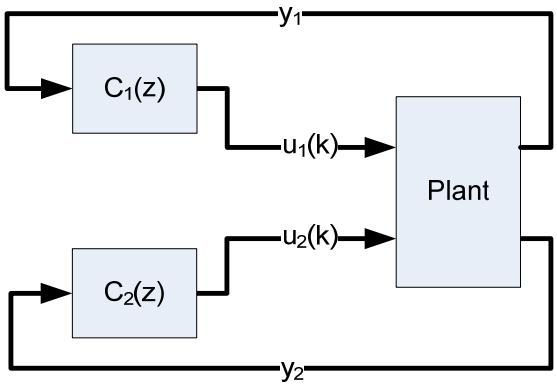


Figure 1.3: Decentralized control. Each controller calculates a control signal at sampling instant k independent of the other.

In decentralized MPC, each subsystem has an individual cost function that the controller seeks to minimize. The interactions with the other subsystems are ignored, although in any MPC optimization disturbances must be accounted for in order to achieve error-free reference tracking. For example, the control designer could use the generalized approach for disturbance modeling in linear MPC presented in [25]. If the

interactions between subsystems are significant, however, decentralized MPC can have the same interaction problems as any other decentralized control approach. Stability guarantees can be difficult to develop, although Magni and Scattolini developed a contractive constraint that provides stability for nonlinear systems [26]. Therefore researchers have developed several techniques for improving the decentralized performance while avoiding many of the difficulties of centralized control, although this is still an area of open research [27].

A step up from decentralized MPC is the approach commonly referred to as *distributed* MPC, first proposed by Jia and Krogh in [28]. In this architecture, each interconnected subsystems has its own controller and individual cost function to optimize. However, the controllers communicate to each other their predicted control inputs and plant outputs (delayed by one time step) as shown in Figure 1.4. This gives each controller a better idea of future disturbances, allowing better prediction and therefore better performance. Stability was guaranteed through the use of a contractive constraint. The same authors also developed a method for bounding the disturbances that each subsystem inflicts on the others via constraints [29]. The technique was further refined by the authors and others in a widely-cited article wherein iteration is added between the subsystems' controllers [30]. Mercangoz and Doyle further developed the distributed MPC controller in [31] by iterating solutions between the controllers based upon a distributed decentralized estimation and control structure, and applied this approach to the standard "quad-tank" experimental system first detailed in [32]. While all of these papers involved breaking the system into subsystems, each with its own

quadratic programming algorithm, in [33] the authors took the approach of splitting the optimization itself up among the controllers, which then iterated to a solution using a feasible direction technique.

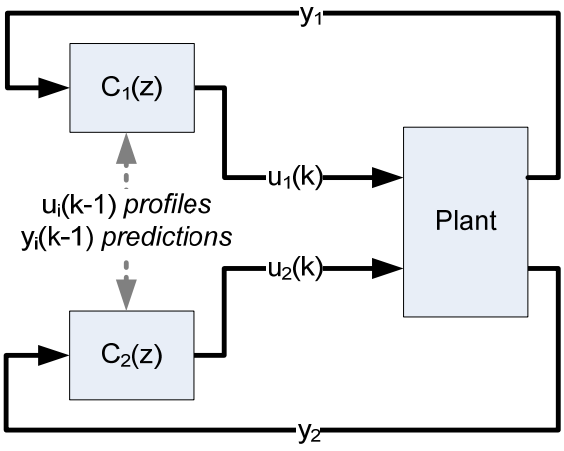


Figure 1.4: Distributed control. Each controller calculates the signal at sampling instant *k* based in part upon what the other controllers calculated at the previous sampling time *k-1*.

Venkat et al. observed that in distributed MPC, where each plant minimizes a different cost function rather than a shared function, the system can approach a condition where the global optimum is not achieved, since the subsystems are each at their own local optima and cannot move away without increasing their own cost functions [18]. This condition is commonly referred to as a Nash equilibrium [34]. The authors suggested an approach where all of the distributed controllers are seeking to optimize the same cost function, naming this feasible cooperation-based MPC (FC-MPC) [35]. This architecture is shown in Figure 1.5. The solutions to each optimization are communicated between controllers and iterated, and the solution is shown to approach

the centralized controller's optimum. Stability analysis, based upon suboptimal MPC theory, is presented in [36]. A detailed treatment of the algorithm can be found in [12], and the algorithm was extended to nonlinear systems in [37]. Other cooperative algorithms with a focus on agent negotiation and game theory can be found in [38, 39]. This approach was further developed to allow setpoint tracking (rather than just regulation) in [40]. Further research efforts refinements involved breaking the plant into subsystems which are grouped into neighborhoods; each controller communicates with its neighbors at every time step while communicating with plants in other neighborhoods less frequently [41].

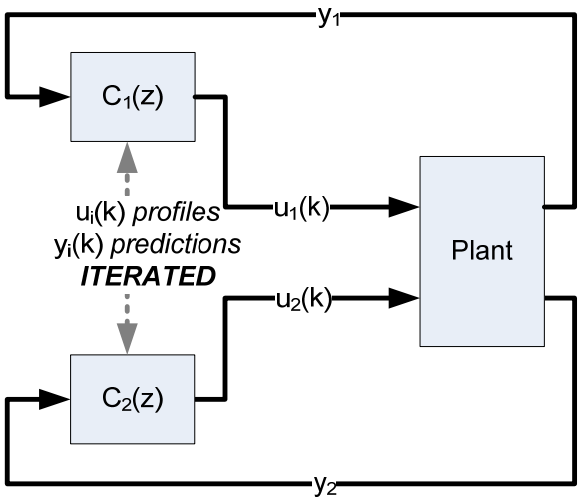


Figure 1.5: Feasible cooperation-based control. At each time step *k* the controllers calculate a control signal and communicate the result with the others. This process is iterated until convergence.

In [35] the same authors also proposed an architecture where each distributed MPC controller has only partial knowledge of the other subsystems' cost functions; they refer to this as partial feasible cooperation-based MPC (pFC-MPC). The authors observe that pFC-MPC is well suited to meeting operational objectives, e.g., tracking a local setpoint, even if the resulting system behavior is not globally optimal. This approach is similar to the modular multivariable controller proposed in [42]. Unlike the approaches where every subsystem communicates with every other, design of the control architecture in this mode requires knowledge of the particular system being controlled.

Most of the preceding work looks at linear plants, although most large processes are inherently nonlinear. The distributed MPC architecture has also been examined in a nonlinear framework as well. Dunbar and Murray applied the architecture to stabilizing the formation of vehicles; this is different from the control systems examined previously in that the cost function is the only thing that connects the system, since the dynamics of the individual actors are not coupled [43]. A "consistency constraint" is used to ensure that the closed loop dynamics do not vary a great deal from the prediction; this ensures stability. The same approach is applied to dynamically coupled systems, in particular a pair of van der Pol oscillators, in [44]. Liu et al. developed a nonlinear DMPC approach based upon Lyapunov control design in [45]. This architecture is also hierarchical in that one controller stabilizes the system and another uses extra available actuators to improve the performance. Another paper on decoupled systems looks at networks of nonlinear systems (e.g., vehicle formations and airport camera networks). In this paper each subsystem is assigned a node, and the controller has knowledge of its own plant as

well as the neighboring plants [46]. Zhu and Henson observe that many large plants can be split into linear and nonlinear elements, and choose this distinction to guide their separation into subsystems; their approach varies from the others presented in that the MPC optimizations are performed sequentially [47].

Another important field of effort in the DMPC arena has been separation of the plants and development of the objective functions. For example, in [48] the authors used the Dantzig-Wolfe decomposition to break up the optimization problem; this decomposition was used to implement a coordinated decentralized MPC. A method for partitioning large systems based upon graph theory and geared towards MPC was reported in [49]; the chosen application for this technique was a drinking water network. In [50] a modeling scheme was developed that treated coupling of the subsystem models as the tuning parameter. Survey articles have begun to collect and categorize the different approaches to DMPC; a comparison of different optimization methods is given in [51], and a recent survey of the state of the art in distributed MPC is presented in [52]. Many of these DMPC approaches require local knowledge of the interaction and plant dynamics of all other subsystems, and do not account for the possibility of different interconnection topologies simplifying the architecture. Regardless of how the plants are interconnected, each controller must communicate with every other one. Additionally, each DMPC controller generally has a model of each plant in the system. If one of the plants is changed or if conditions change so that the linearization (if any) is no longer valid, every controller in the network must be changed as well.

This creates the potential for research into leveraging particular network topologies into more effective, specialized control architectures, since in a typical HVAC application there are hundreds of controllers spread out over a very large physical area. Progress is currently being made in this area. For example, to address the problem of serially connected systems, a DMPC scheme specially constructed for this topology was presented in [53]. To reduce the communication load between controllers, a non-cooperative algorithm that only requires neighbor-to-neighbor communication was presented in [54].

A recent paper proposed an approach that used the sensitivities of the local cost functions to the actions of their neighbors in computing local controller actions [55]. This required that each subsystem have knowledge of its neighbors' dynamics, however, since the sensitivity was calculated with respect to the neighbors' inputs.

Research into the area of decentralized MPC in all its varieties is still an active field. As such, the terminology used in each of the cited papers is not always the same, and can lead to some confusion when surveying the literature. Surveys of this field of research, including proposed classification schemes, can be found in [27, 56].

MPC & HVAC

Many researchers have used MPC-based approaches to control airflow in the air handling units of large building cooling systems. These approaches have either used an MPC algorithm to directly control the system actuators as in [57], or used an MPC algorithm to set gains for local PID controllers as in [58] and [59]. Another approach is

to use an MPC algorithm to perform controller self-tuning, whether for commissioning or to adapt to changing conditions, as done by MacArthur in [60] and Dexter in [61]. Sousa et al. used MPC with fuzzy logic to control water valves in radiative heating systems [62]. He et al. extended this fuzzy model approach to the control of an HVAC air handling unit (AHU), such as that found in a large building HVAC system [57]. Xi and Poo developed a learning algorithm, combined with nonlinear MPC, to control a variable air volume (VAV) AHU [63]. Yuan and Perez used MIMO MPC to control a VAV AHU without the usual "controller sequencing" frequently used in industrial practice [64]. Finally, Huang et al published a series of papers applying LMI-based robust MPC to VAV control [65-68].

Given that the disturbances that HVAC systems endure are somewhat predictable (e.g., weather changes, outside temperature, building occupancy), MPC is particularly well suited to take advantage of this knowledge. Indeed, given that MPC is an optimization algorithm, it is also well suited to improve energy efficiency while meeting comfort demands. For example, Paris et al used an MPC controller to augment a standard PID controller; this MPC controller had knowledge of predicted weather data and adjusted the PID controller's output accordingly, improving energy efficiency and disturbance rejection [69]. Privara uses weather data as well, but used a MIMO MPC controller to control the building heating system explicitly, improving efficiency over the standard PID approach [70]. This paper also proposed some system identification techniques well suited to this application. May-Ostendorp et al used MPC to control the window openings in a mixed-mode building to reduce energy consumption [71]. Frieire

et al developed an MPC controller that balances energy consumption with predicted mean vote (PMV), a widely used occupant comfort measurement [72].

Some work has been done on controlling the refrigeration cycle itself with MPC. Leducq et al. applied a MIMO MPC algorithm to a single-evaporator water chiller system using a nonlinear model, controlling cooling and temperatures using compressor speed and water flow [73]. Changenet et al. developed a modeling technique to build an MPC controller that uses the EEV to regulate superheat, and in a later paper adapted this technique to the control the condenser pressure, thus minimizing the disturbances the superheat controller need to reject [74, 75].

Distributed MPC is also beginning to be applied to HVAC problems, including building systems control [76]. Morosan et al explored muti-zone building heating using a DMPC approach, and comparing it with centralized MPC [77]. An interesting facet of this work is the incorporation of building occupancy expectations into the MPC algorithm. The same authors decomposed the DMPC problem into a linear programming algorithm for a similar control application in [78]. Additionally, the application of DMPC to a multi-case refrigeration system was presented in [79]. Given the complex nature of HVAC systems, the application of distributed MPC promises to be a fruitful research area.

Multiple Evaporator Systems

This research will involve a variable refrigerant flow (VRF) system with multiple evaporators. These systems have not been implemented widely in the United States,

despite their popularity elsewhere and their potential for energy savings over traditional water loop based configurations [80]. VRF systems carry their own set of controls challenges; due to the cross-coupled dynamics of VCC systems, single-input single-output (SISO) control approaches are severely limited, as shown in [81]. A simulation comparison was performed by Aynur, showing that a VRF system can give significant energy efficiency improvements over a VAV system in the same building; these ranged from 27% to 57% improvement for the building studied [82]. Previous research efforts from the Thermo-Fluids Control Laboratory at Texas A&M proposed a hierarchical MPC approach borrowed from the chemical industry, where a supervisory controller selected cooling and pressure setpoints for local MPC controllers to meet; the particularities of VCC systems were leveraged to improve the controller design [83, 84]. In their research into multiple-evaporator systems, Choi and Kim showed that using the EEVs and compressor in combination to modulate the capacity of each evaporator can result in better operating efficiencies, although each EEV has a strong effect on the other evaporator [85]. Park et al. also explored this combination, calculating optimum EEV for a given compressor speed and load ratio [86]. In neither case is the design of a dynamic controller considered, although they reference the work of Fujita, who investigated using a PID loop with an EEV and variable speed compressor to control superheat and cooling capacity in a multi-evaporator system. Asada and He explored using feedback linearization in a PI loop controlling the compressor speed with a multiple-evaporator system, although simulation results only were presented [87]. Kim et al. investigated using MPC to control a multiple evaporator heat pump; in this case,

the EEV was used to control the evaporator temperature. This method was shown to have performance improvements over using PI loops [88]. Shah et al. performed modeling work of a multi-evaporator system and explored control techniques; as expected, a MIMO approach was found to be superior to SISO techniques for control, due to the complexity of system dynamics [89]. Chiou et al. applied fuzzy control techniques to energy savings for small scale household air conditioners with multiple evaporators, and found better temperature regulation than could be had with traditional on/off techniques [90]. A recent survey of the various VRF configurations currently used and their benefits is presented in [91]. Wang provides a survey of supervisory and optimal control techniques used in the HVAC industry [92].

A Primer on Vapor Compression Cycles

Much of the research in this dissertation is applied to multiple-evaporator VCC systems. In typical building systems in the United States, chilled water or air is ducted to various areas to perform the required air conditioning. In contrast, VRF systems, which are widely used in Europe and Asia, use a variable speed compressor (or bank of compressors) to deliver the refrigerant directly to evaporators servicing the different areas. The cooling delivered to each zone can be varied based upon its specific cooling needs. This approach has the potential for energy efficiency improvements by avoiding losses associated with extensive ducting, and by using variable speed compressors and fans to deliver only the cooling required. Multi-evaporator systems also find application in supermarket refrigeration, where each cooling case has its own evaporator and can be regulated to the appropriate temperature for different foodstuffs—milk, frozen goods, etc. Refrigerated trucks also use this approach for similar reasons. Luxury automobiles can also use this technology so that each passenger can adjust the temperature to improve his individual comfort.

Figure 1.6 is a schematic of an example system with *n* evaporators. The pressure-enthalpy (P-h) diagram for a two-evaporator system is given in Figure 1.7. Each region with a separate evaporator is designated as a zone; regulation of the various zones' temperatures is the main control objective. These zones are generally subject to heat disturbances that tend to increase the temperature of the zone. Examples include heat generated by computer equipment in a server room, opening a milk case door, or an increase in the outside air temperature. Heat is removed from the zone by the VCC

process; this heat transfer is referred to as cooling. The working fluid used in the cycle is a refrigerant, e.g. R134a.

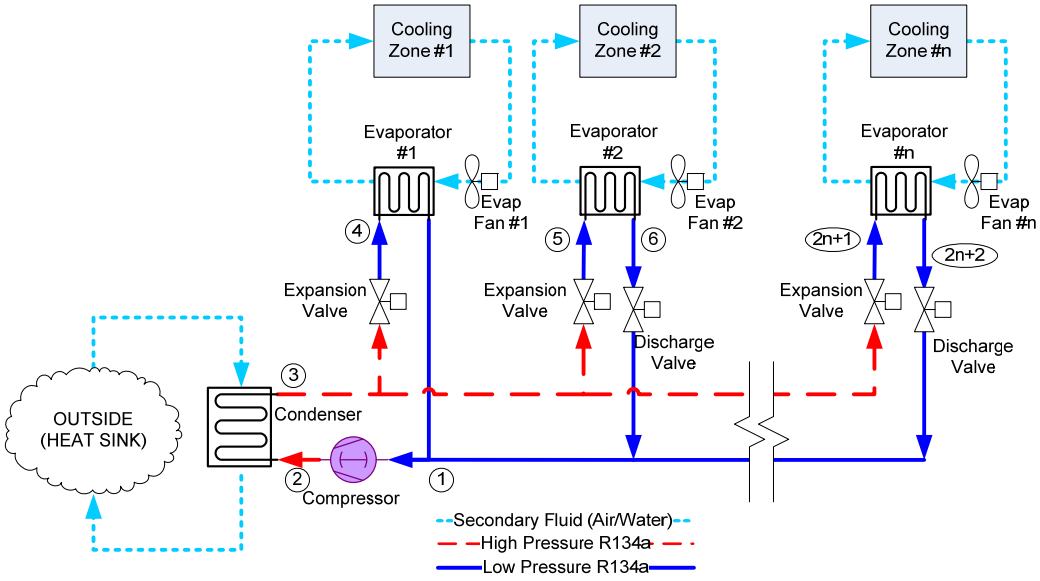


Figure 1.6: Schematic of a multiple evaporator vapor compression system.

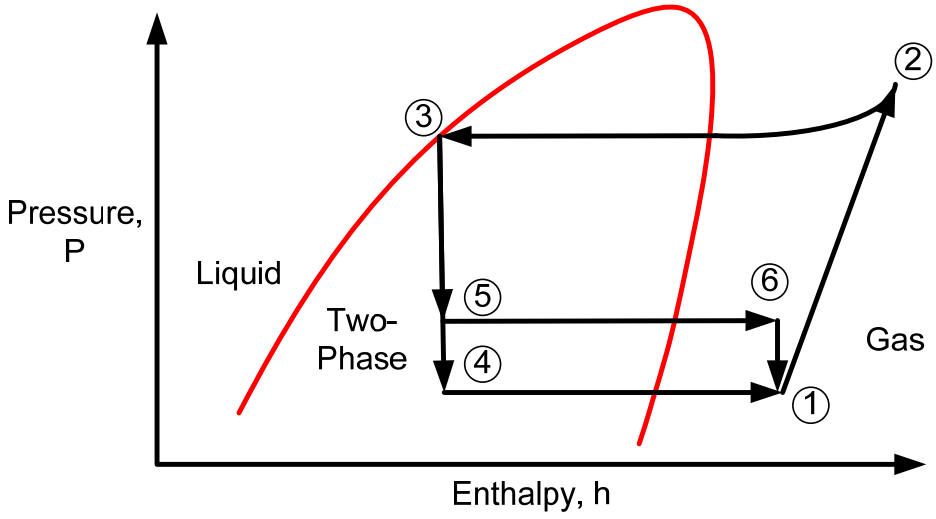


Figure 1.7: Pressure-enthalpy diagram for two evaporator system

The cycle progresses as follows. Low-temperature, low-pressure refrigerant vapor, carrying the heat energy absorbed from the zones by the evaporators, is compressed to a high pressure, high temperature vapor (process 1-2). The refrigerant then passes through a condenser, where heat energy is rejected to the outside air (process 2-3). A condenser fan is used to move air across the condenser to aid in this process. This high-pressure liquid refrigerant is then distributed to each of the expansion valves, which meter the refrigerant to the evaporators. In the process of metering, the refrigerant is expanded to a low-pressure, low-temperature two-phase fluid (processes 3-4, 3-5). As refrigerant passes through the evaporator, it absorbs heat from the zone and evaporates to a low-pressure, low-temperature vapor; an evaporator fan moves the air to aid in the heat transfer process. Additionally, a discharge valve can be placed at the end of any evaporator to allow operation at higher pressures than others, e.g., to cool vegetables with one evaporator and frozen foods with another.

If the evaporator is operating correctly, the refrigerant will be completely evaporated, and the temperature at the evaporator exit will be higher than the entrance; this temperature difference is referred to as superheat, and is an important condition in VCC control. Another signal of note is the rate at which the evaporator removes heat from the zone; this is referred to as cooling, measured in kW. In practice, evaporator superheat is measured as a temperature differential (Equation 1.1) and cooling on the refrigerant side of the heat exchanger (Equation 1.2) is calculated based on temperature measurements and a refrigerant mass flow rate that is measured directly or calculated

from temperature sensors and manufacturer data. The terms used in these two equations are defined in Table 1.1.

$$ {}^{n}sh = {}^{n}T_{ro} - {}^{n}T_{sat} \tag{1.1} $$

$$ {}^{n}q = {}^{n}\dot{m}_{ref}\left({}^{n}h_{o} - {}^{n}h_{i}\right) \tag{1.2} $$

Table 1.1: Superheat and Cooling Nomenclature

${}^{n}sh$	Superheat of the n^{th} evaporator, °C
${}^{n}T_{ro}$	R134a outlet temperature of n^{th} evaporator, °C
${}^{n}T_{sat}$	R134a saturation temperature at n^{th} evaporator pressure, °C
${}^{n}q$	Cooling at the n^{th} evaporator, kW
${}^{n}\dot{m}_{ref}$	Mass flow of R134a for n^{th} evaporator, kg/sec
${}^{n}h_{o}$	Enthalpy of R134a at n^{th} evaporator exit, kJ/kg
${}^{n}h_{i}$	Enthalpy of R134a at n^{th} evaporator inlet, kJ/kg

CHAPTER II

EXPERIMENTAL APPARATUS

Overview

Justification and Organization

The experimental system used for the research presented in this dissertation is a custom-built, small-scale three-evaporator water chiller. This system is intended for dynamic model validations, control development, fault detection, and other research for the Thermo-Fluids Control Laboratory. These uses and the desire for modularity and flexibility were the primary considerations during the process of component selection and system construction. This chapter first gives an overview of the flow logic and construction of the primary (refrigerant) system and the secondary (water) system. Details of the individual components for the primary and secondary loops are presented. The sensors used in the system are described. Finally, the software and data acquisition systems are detailed. Wiring schematics for the electrical and electronic components can be found in the Appendix. Figure 2.1 is a photo of the entire system.

Figure 2.1: Experimental apparatus

Primary System

The refrigerant or "primary" side of the system uses R-134a as the working fluid, which flows through copper tubing. Where possible, Swagelok brand compression fittings were used to join the tubing together, since they are reusable (although new ferrules must generally be purchased) and are less prone to leakage than SAE 45° flare fittings. For some components, such as at the compressor inlet and outlet ports, soldered connections were required. In general 1/4" tubing is used for lines carrying liquid, and 3/8" or 1/2" tubing is used for gas lines and lines carrying two-phase fluid (i.e., from the expansion valves to the evaporators). A liquid receiver is installed at the end of the condenser to ensure that saturated liquid is fed to the expansion valves. There is a bypass manifold for the receiver so that tests with refrigerant subcooling can be conducted if desired. Manual shutoff valves are used throughout the system to reconfigure the refrigerant flow as desired. These valves also allow retention of refrigerant while sections of the system are being worked upon. Vapor compression is provided by a variable speed compressor, and the refrigerant expansion is controlled with electronic expansion valves. The heat exchangers (condenser and evaporators) are all shell-in-tube style that use water as the secondary fluid. Figure 2.2 is a schematic of the primary side of the system. Component descriptions are provided later in the chapter; additionally, the Appendix contains a complete table of the schematic reference numbers (A2, MV8, etc.) and a description of the role played by each.

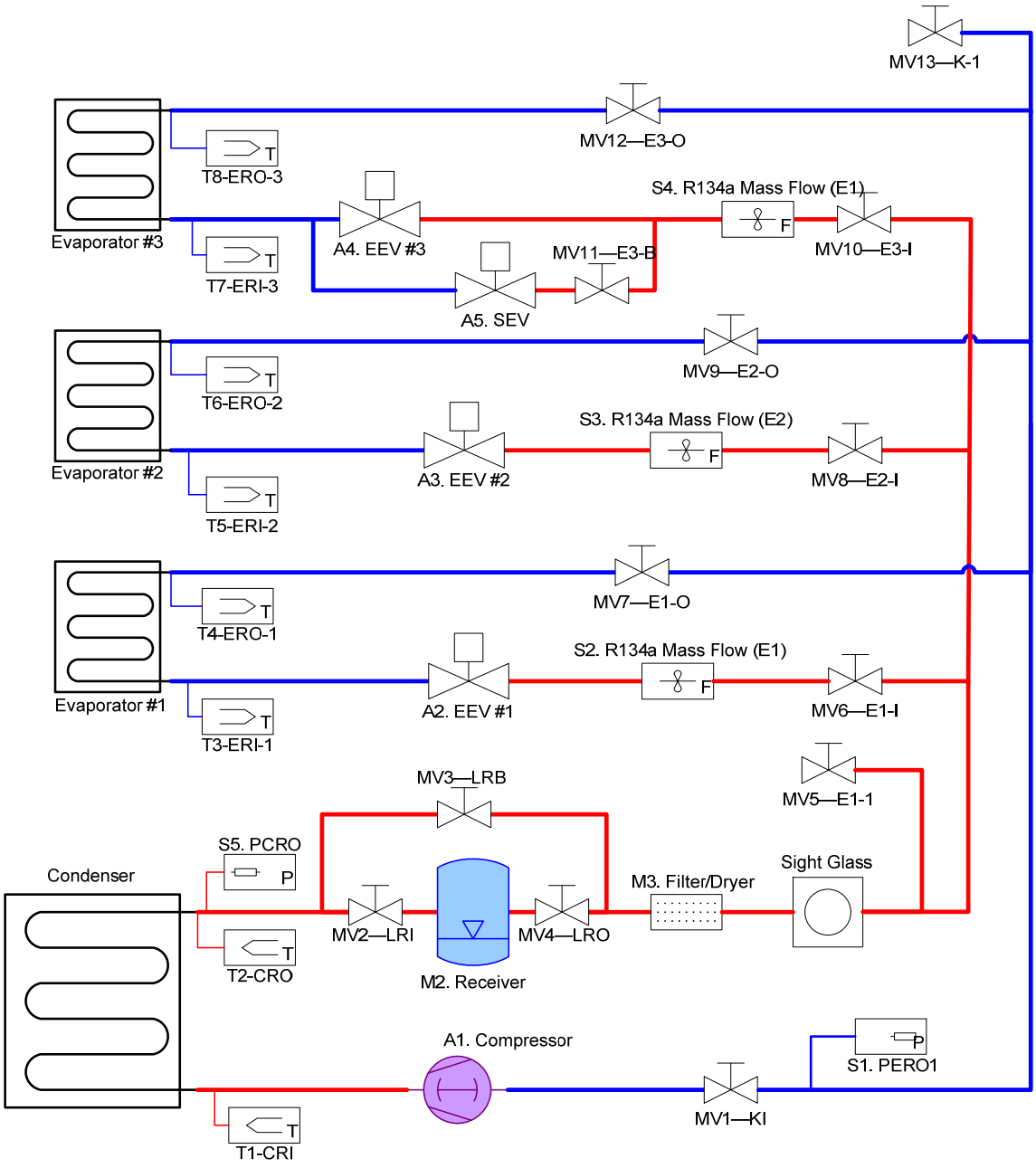


Figure 2.2: Primary (refrigerant) loop

Secondary System

Water is used as the "secondary" working fluid in the heat exchangers. There are three 50 liter water tanks that represent separate cooled zones. Water from these tanks is pumped separately to the appropriate evaporator and back to the zone, just like in a real air conditioner. A tank system containing 260 liters of water serves as the heat sink, representing the outside air of an air conditioning system; this water is fed into the condenser. Figure 2.3 is a schematic of these components. There is a separate water chiller loop that is tied into the building's chilled water supply so that thermal energy can be removed as needed from the condenser supply water.

Since each of the heat exchangers has its own isolated water supply, temperature control of a set of rooms or zones can be simulated. Additionally, there are a set of individually controlled pumps that pump water from the condenser tank into each of the evaporator tanks. This allows for simulating disturbances due to outside air temperature, leaky windows, etc. Use of this disturbance can also be used to keep the water temperature in the evaporator tank constant despite the cooling effect of the evaporator; this is useful when carrying out system identification experiments on the VCC system. Since the amount of water in the evaporator tank must remain constant, overflow lines allow excess water to be returned to the condenser tank. Additional disturbances can be created with immersion water heaters; this simulates internal loads such as people, electronic equipment, lights, etc. Figure 2.4 shows a schematic of the overflow disturbance system; photos are shown in Figure 2.5.

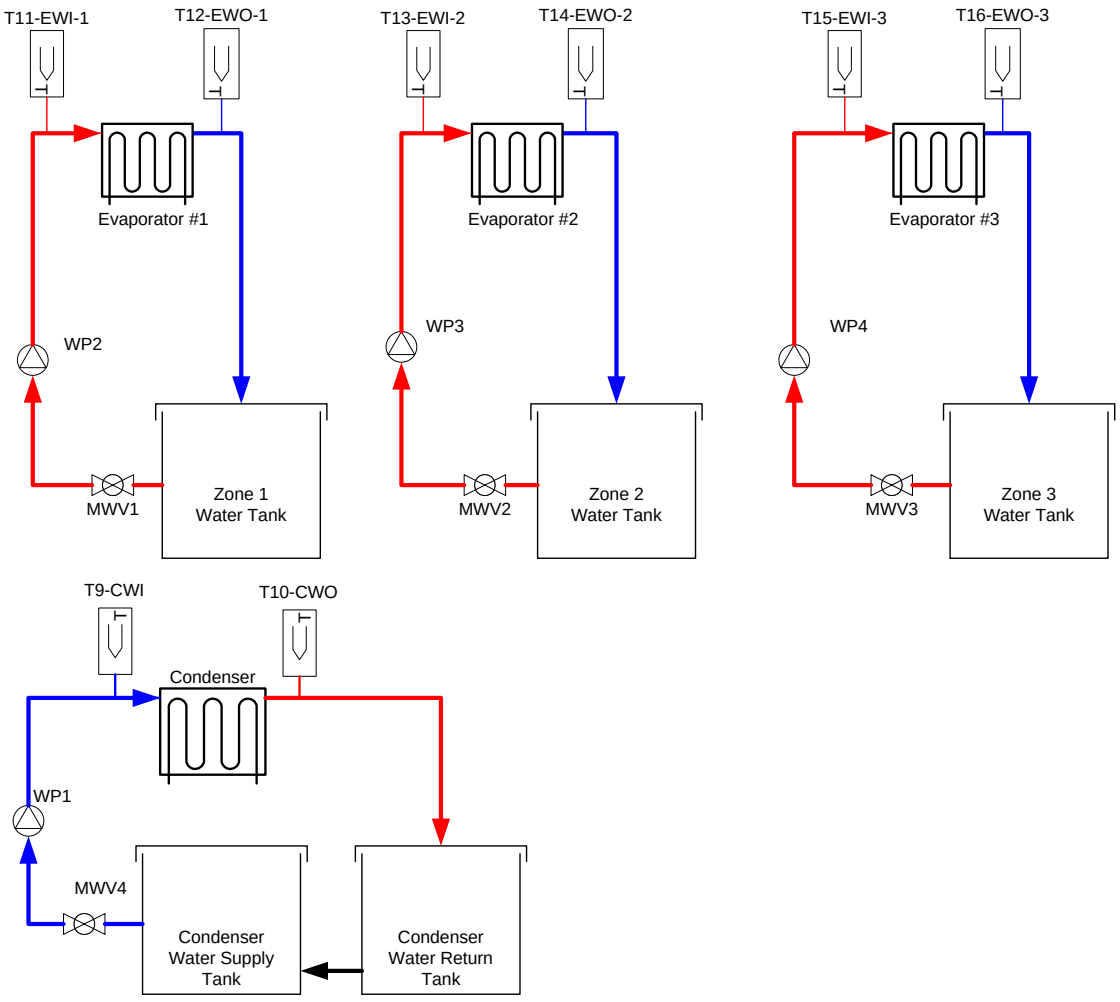


Figure 2.3: Secondary loop schematic

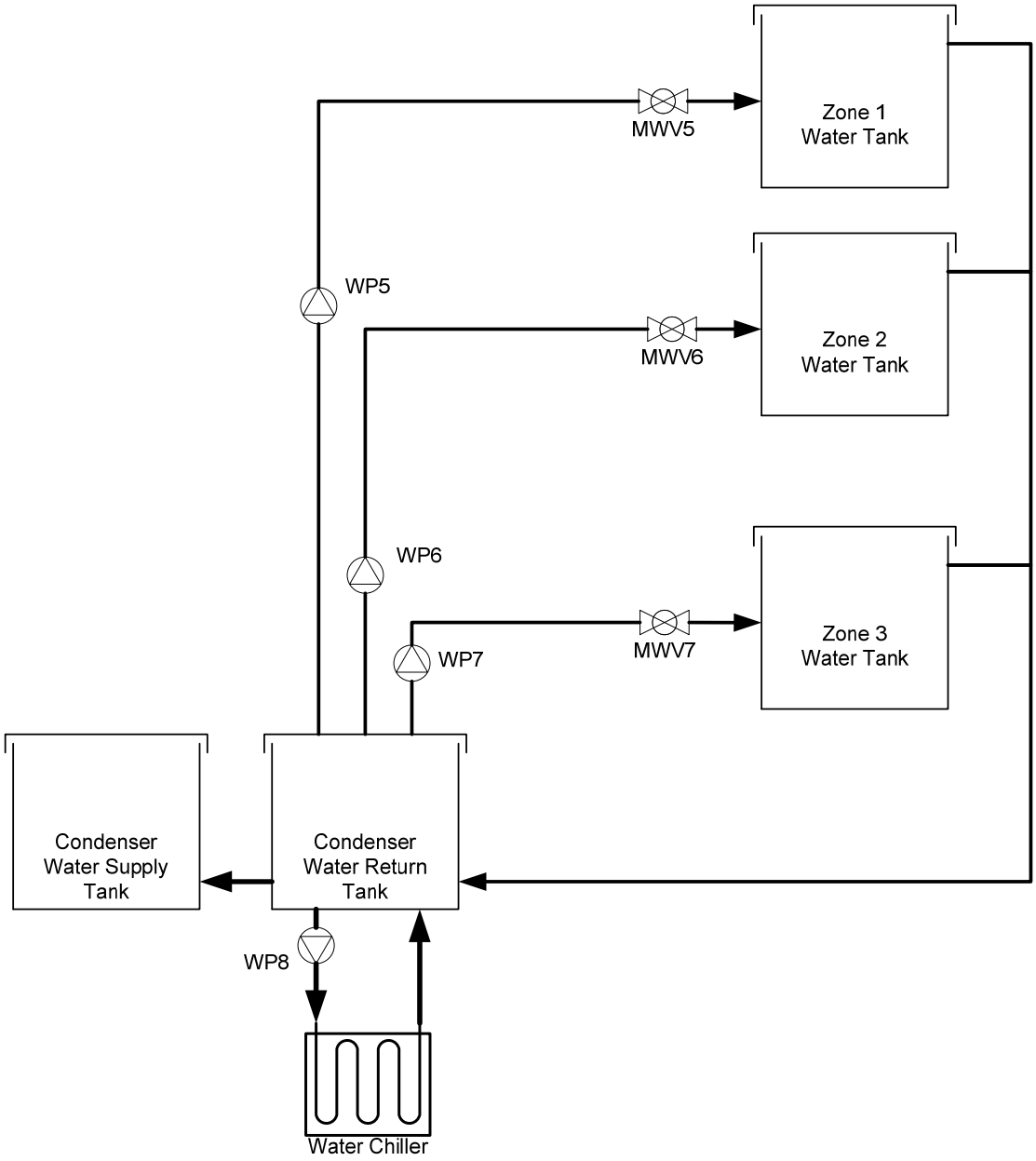


Figure 2.4: Disturbances and heat rejection system schematic

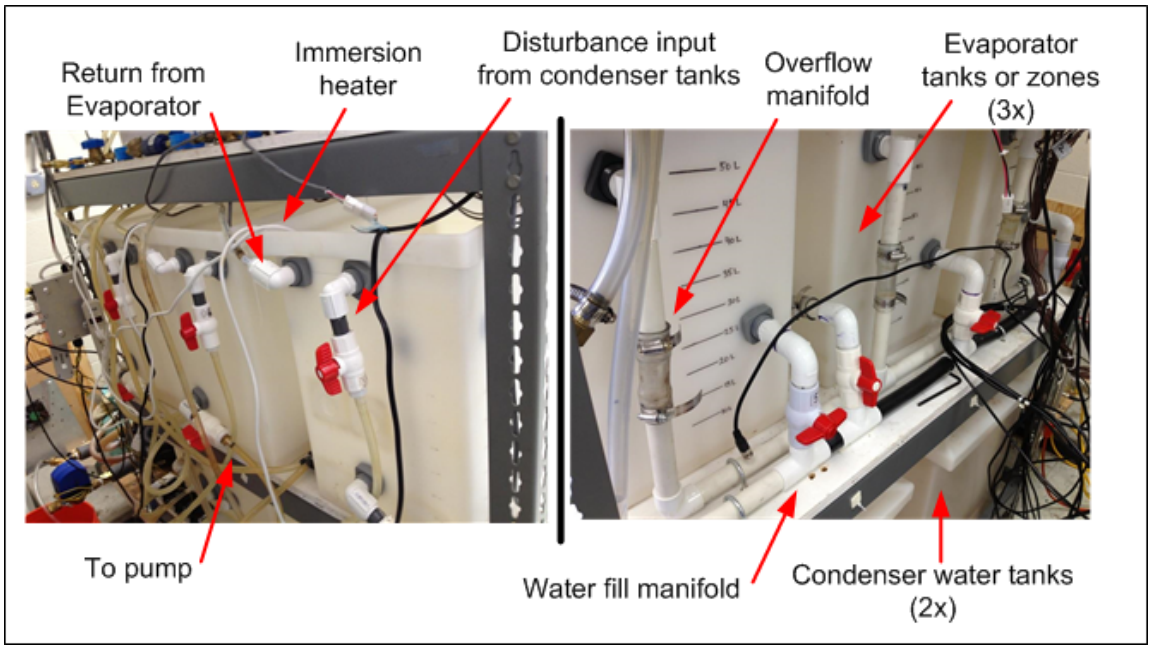


Figure 2.5: Water tank photo

Primary Loop Components

Heat Exchangers

The four heat exchangers—three evaporators and one condenser—installed in the system provide the interface between the primary and secondary loops and are the means by which thermal energy is transferred within the system. These heat exchangers are all shell-in-tube style heat exchangers, manufactured by Packless Industries. Refrigerant passes through the inner loop, made of copper, while the water being chilled or heated passes through an outer loop made of steel tubing. The condenser is rated for 1.5 ton systems, and the evaporators are rated for ½ ton apiece. Figure 2.6 shows photos of the evaporators and of the condenser.

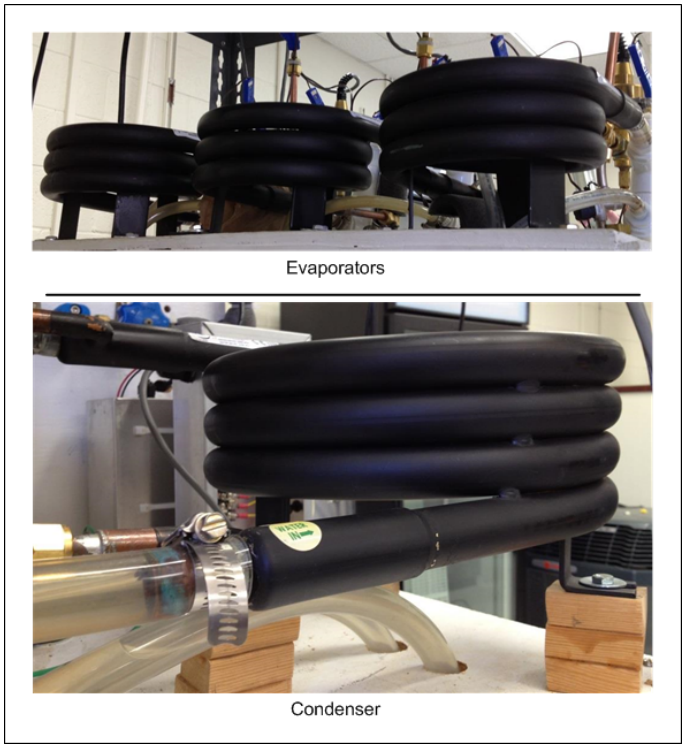


Figure 2.6: Heat exchangers

Refrigerant Expansion Valves

The expansion valves used in the system are electric expansion valves (EEVs) manufactured by Sporlan. Each valve is controlled by a Sporlan IB interface board. These boards accept a 4-20 milliamps (mA) command signal from the user and position the valve proportionally according to the signal using the step motor integral to the valve assembly. They have a resolution of 1596 steps at 200 steps per second, for a total travel time of approximately 8 seconds from completely open to completely closed. Figure 2.7 is a photo of the electronic expansion valves. Wiring details can be found in the Appendix.

In addition to the EEVs, the third evaporator has an auxiliary selectable expansion device for model validation or other experimental purposes not related to the research presented in this thesis. This valve is a prototype Silicon Expansion Valve (SEV) manufactured by Microstaq. In order to use this valve, a bypass valve on the inside of the evaporator support structure must be opened and the EEV on evaporator 3 completely shut. This valve is currently not wired into the system, although similar valves were used in the data taken in Chapter III. Figure 2.8 shows a photo of the valve as well as the bypass line.

Figure 2.7: Electronic expansion valves (EEVs)

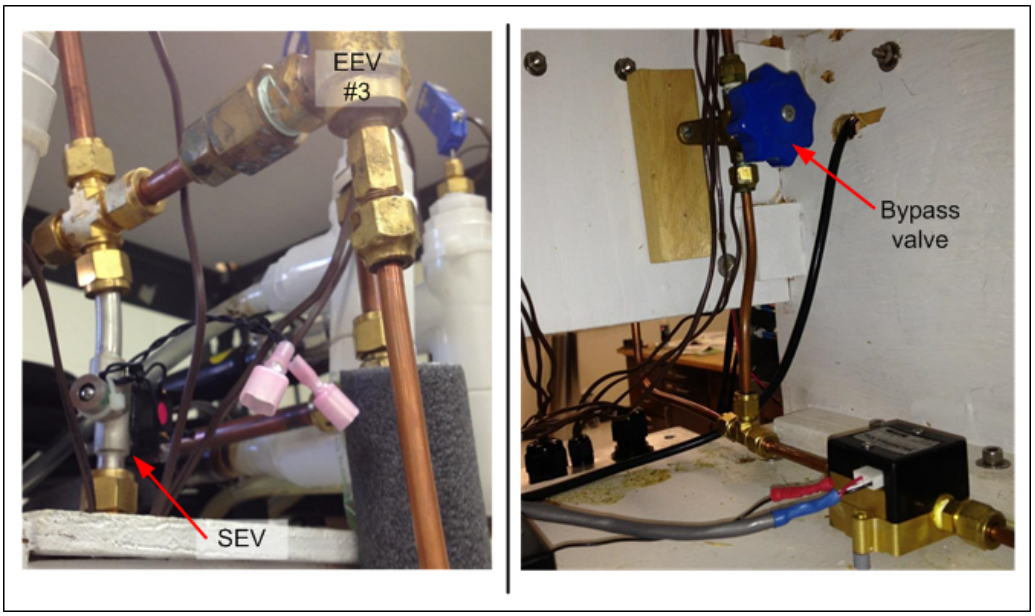


Figure 2.8: Auxiliary expansion device manifold

Compressor

The compressor for the system is a Sierra model manufactured by Masterflux. This is a scroll-type variable speed compressor that uses a 48V DC power supply and is designed to operate with R134a refrigerant. This voltage is fed into a compressor control module included by the manufacturer, which accepts control signals from the user and regulates the compressor speed. The control module allows for a manual switch to turn the power on and off, outputs a tachometer signal, the current consumed, and accepts a 0-5 volt signal to control compressor speed. The compressor speed varies from 1800 to 6500 RPM. Total compressor capacity is approximately 1.5 tons of cooling.

Miscellaneous Safety Components

A liquid line receiver made by Henry Technologies is placed at the end of the condenser. This receiver serves as a safety measure to ensure that only liquid is fed into the expansion valves, thus decreasing the risk of choking the valves. Using a receiver also ensures that the liquid will be saturated rather than subcooled, which enables some assumptions with regards to enthalpy calculations and system conditions. For some experiments, a liquid receiver is not desirable; therefore, a bypass manifold consisting of manual shutoff valves is also included. These manual shutoff valves are also installed at various places in the system; this allows one of the evaporators to be shut off, or for a section of the primary loop to be closed off while another section is being worked on. This way the majority of the refrigerant in the system can be stored in the liquid receiver while the system is being worked on, resulting in a minimum of refrigerant loss. Figure 2.9 is a photo that displays the compressor, receiver, and shutoff valves.

As an additional safety measure, a filter/drier is placed in the refrigerant flow stream after the liquid receiver manifold in order to protect the expansion devices. In order to provide real-time visual information to the researcher running the system, a sight glass is also located here to allow the operator to verify that only liquid refrigerant is being passed to the valves. Figure 2.10 is a photo of these components.

Figure 2.9: Photo of compressor, liquid receiver, and manual shutoff valve

Figure 2.10: Filter drier with sight glass

Secondary Loop Components

Water Pumps

The water flow for each evaporator is controlled by a DC-powered water pump manufactured by Swiftech for water-cooled computers. These pumps can pump up to 1.5 gallons per minute (gpm) at 12V, and the flow can be varied by varying the supply voltage. An amplifier circuit was built for each pump; this circuit accepts a 4-20 mA control signal from a DAQ system and converts it to a 6-12 VDC power signal.

The same model pump is used to pump warm condenser tank water into the evaporator tank. This might be done to model outside air disturbances or regulate evaporator tank water temperature. These pumps are controlled digitally (on/off) and are set up to run at full speed only. The condenser pump is a larger model than the evaporator pumps, and is run constantly with the system. Later system modifications might include building an amplifier control circuit for this pump so that a variable speed condenser fan can be simulated. In addition, ¾" PVC ball valves are used throughout as shutoff valves for the water supply. Figure 2.11 includes photos of the water pumps.

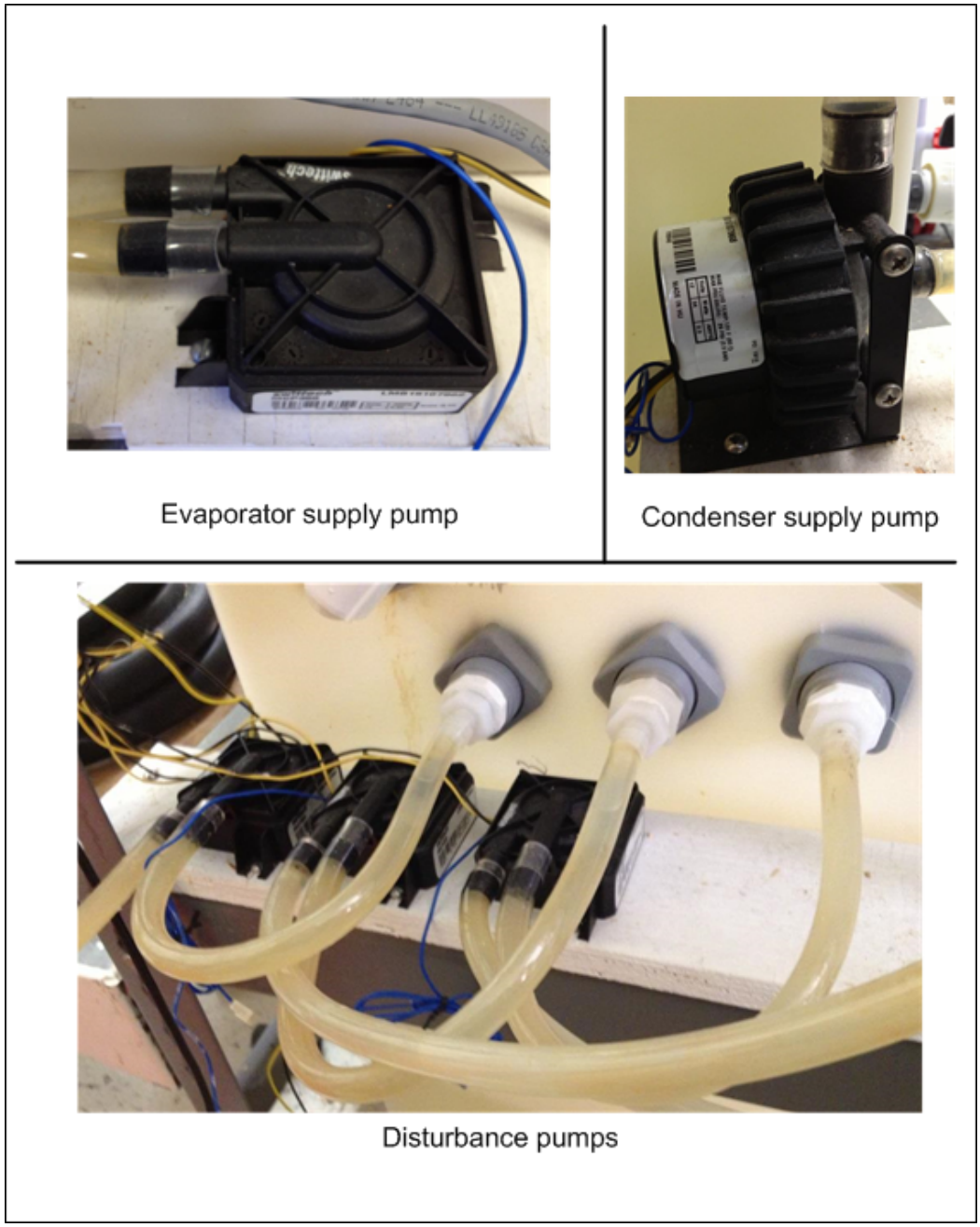


Figure 2.11: Water pump photos

A final DC pump is used to move water through an outside heat exchanger to dump heat outside of the system. This heat exchanger is connected to the chilled water supply of the building. An AC pump is used in series with the chilled water line to provide the necessary boost to the chilled water supply. Both of these pumps are controlled digitally by the DAQ system to keep the condenser water supply temperature constant, usually at 30° C for the tests displayed in this thesis. This arrangement is shown in Figure 2.12.

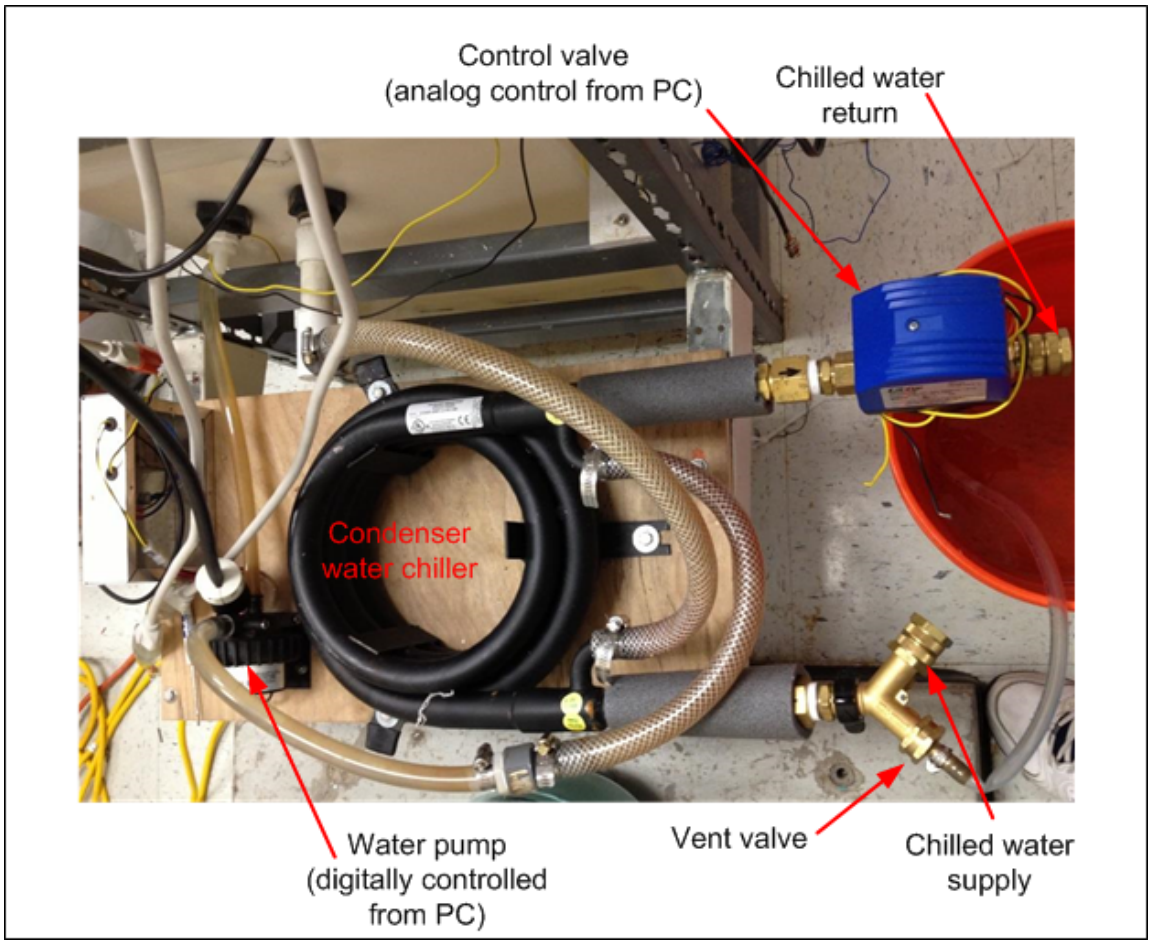


Figure 2.12: Condenser water chiller

Transducers

Thermocouples

Temperature measurements are made with type T thermocouples immersed in the tested fluid. These thermocouples have ungrounded sealed tips and are of the low-noise variety. The thermocouples are immersed in the measured fluid; a Swagelok tube fitting grips the shaft of the thermocouple, sealing the measured fluid from the air. The thermocouples are wired into a thermocouple terminal board with built-in cold junction compensation; this board is connected to a PCI thermocouple board on the computer.

Pressure Transducers

Pressure measurements of the refrigerant are made using sealed stainless steel diaphragm-type pressure transducers manufactured by Cole-Parmer. A transducer with maximum pressure of 300 psi is used to measure pressure at the outlet of the condenser, and a transducer with maximum pressure rating of 100 psi is used at the inlet of the compressor (suction pressure). All of these transducers output a 1-5 V signal proportional to the pressure, which is fed into the data acquisition boards without processing.

Refrigerant Flow

The transducers used to measure refrigerant flow are volumetric turbine-style flowmeters manufactured by McMillan. These transducers output a 0-5V signal, which is fed directly into the data acquisition board without signal conditioning. Photos of the three refrigerant-related transducers are shown in Figure 2.13.

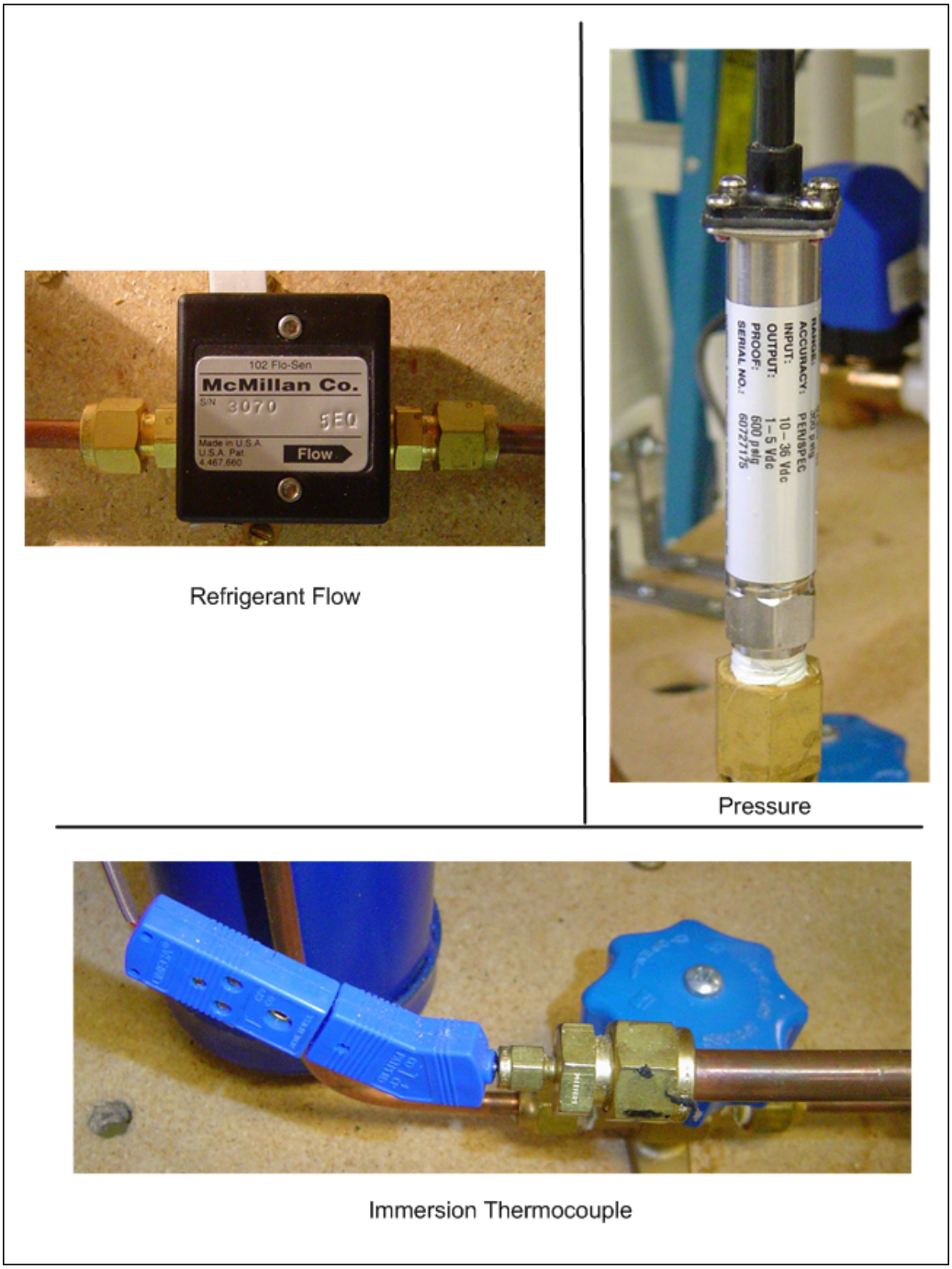


Figure 2.13 Refrigerant cycle transducers

Current Transducer

The current consumed by the compressor is measured with a current transducer made by CR Magnetics. This transducer uses the Hall Effect to measure DC current passing through the wire to the compressor electronics; it outputs a 0-5V signal proportional to the current. Figure 2.14 shows the location of this transducer.

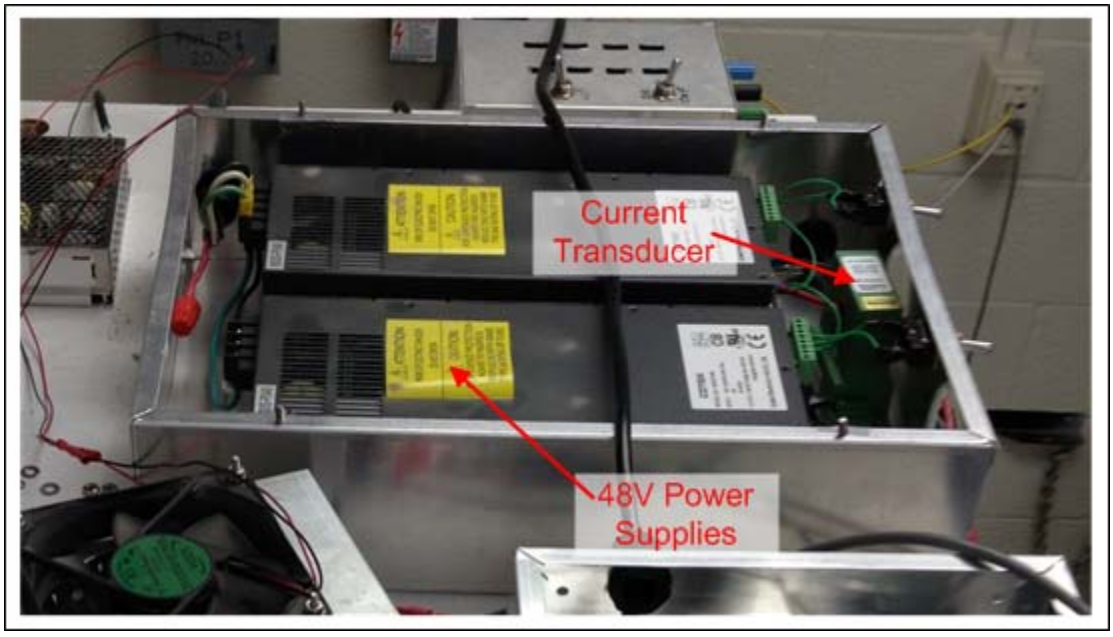


Figure 2.14: Current transducer and 48VDC power supplies

Power Components

The compressor is powered by two 800 Watt DC power supplies. These power supplies accept a 208VAC supply and output up to 16 amps at 48 VDC each. They are connected in parallel to deliver the power needed to run the compressor and are individually switched. They are indicated in Figures 2.14 and 2.15.

Power supplies for the transducers and signal conditioning equipment are also used. A 24VDC power supply is used to power the pressure, refrigerant flow, and compressor current transducers. Multiple 12 VDC power supplies are used to power the water pumps and cooling fans for the system power electronics. A 5V DC power supply is used to power the signal conditioning modules.

Each Sporlan Interface Board (IB)—one for each EEV—is powered by an individual 24 VAC transformer. Per the manufacturer's recommendation, metal oxide varistors (MOVs) are placed across the 120V AC power supply (from the mains) and ground to protect the boards. As an additional measure of safety, fuses are installed on the primary windings of the transformers. Wiring schematics for the EEVs and for all components are included in the Appendix, as well as tabulated data on each component.

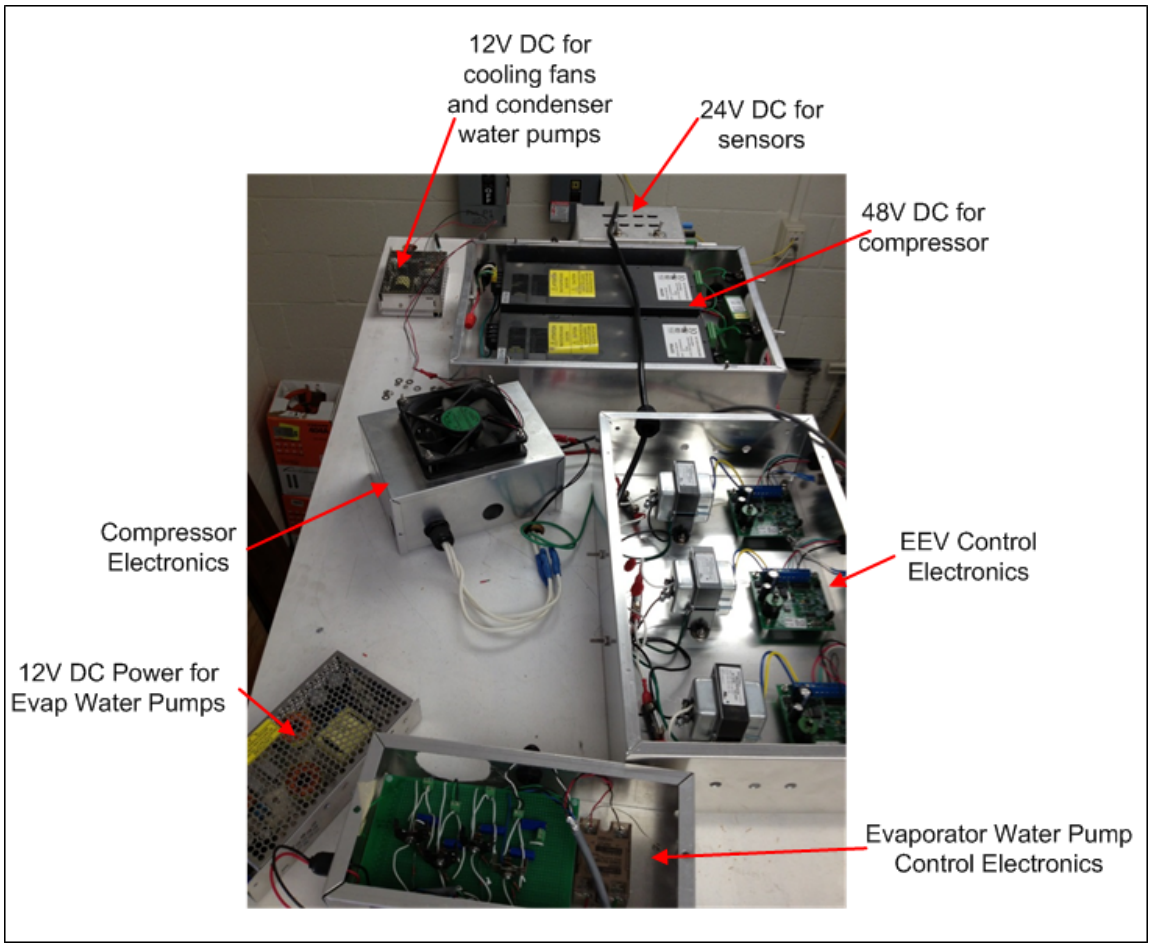


Figure 2.15: Location of power electronics.

Data Acquisition

The data acquisition (DAQ) system consists of three on-board DAQ boards installed on a PC. Temperature measurements are performed using the type T thermocouples detailed earlier and recorded and logged using a Measurement Computing thermocouple board, model PCI-DAS-TC. This board includes cold junction compensation and is capable of handling sixteen thermocouples. For this system arrangement, all sixteen thermocouples are used. Analog output signals to control compressor speed, valve positions, et cetera, are output by a Measurement Computing PCI-DDA-08 board. This is a 12-bit board that has up to eight channels of analog output. Other sensor measurements (pressure, refrigerant flow, and current) are logged using a National Instruments E-Series board, model number E-6023. This board has eight channels when connected in differential mode. It also has eight channels of digital output and two channels of analog output.

An Analog Devices signal conditioning backplane is also used in the system. All analog output channels are processed through optical isolators. These isolators serve the dual purpose of separating the analog output board from the high-powered system electronics, thus protecting the PC, and they also convert the PC output signals into the appropriate signals for the various actuators (e.g., 0-5 VDC to 4-20 ma for the EEVs). A photo of the signal conditioning components is shown in Figure 2.16.

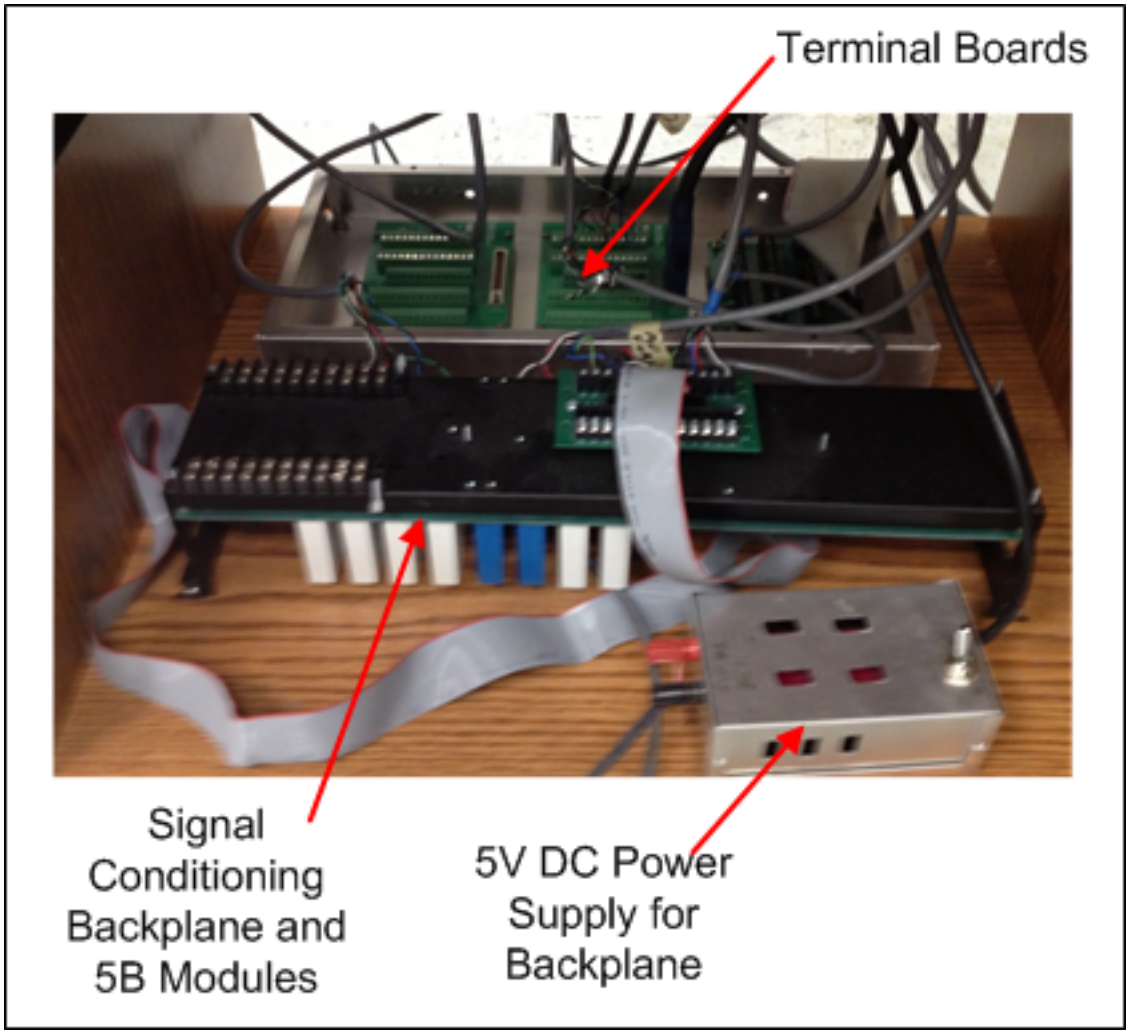


Figure 2.16: Signal conditioning equipment.

Software

The data logging and control functions are performed with WinCon 5.0, a software package by Quanser that provides a convenient interface with MatLab and Simulink. A Simulink model is created and compiled into a program that WinCon executes in real time. Additionally, gains and many other parameters can be varied while the program is running, which is very convenient when tuning controllers or performing experiments. Unfortunately, the selection of available DAQ boards that WinCon supports is very limited. A set of drivers for the thermocouple board and the analog output board was developed by the Alleyne Research Group at the University of Illinois at Urbana-Champaign; these drivers were modified and implemented for the experimental PC. The analog input boards are supported by WinCon, so no driver development was required for these boards. Later upgrades to the experimental system should likely include an upgrade to Quanser's Quarc program, which offers greater flexibility and ease of use when compared to WinCon.

CHAPTER III

CASCADED CONTROL OF EVAPORATOR SUPERHEAT

The Introduction to this dissertation presented the physics of superheat. Examining superheat control in terms of the signals available for measurement and their relative dynamics leads to the new control architecture presented in this chapter.

As previously discussed, the difference between the refrigerant temperatures at the inlet and outlet of the evaporator is referred to as evaporator superheat. In standard superheat control techniques, this difference is the controlled output, and is regulated with an expansion valve that meters the refrigerant into the evaporator. This valve can have mechanical feedback, such as the thermostatic expansion valve. This work uses electronic feedback and control with the electronic expansion valve (EEV). A high level of superheat is generally undesirable, since this condition implies the evaporator is being starved and not being used to its full capacity. Also, the high superheat means that more of the heat transfer is occurring in a gas-to-gas manner, which is less efficient than heat transfer through evaporation. Conversely, too low a level of superheat runs the risk of flooding the evaporator and passing liquid into the compressor. Thus a balance between efficiency and safety must be struck through effective regulation.

Examination of the behavior of the two temperatures being compared in the superheat calculation suggests that they can be controlled independently of each other. For example, Figure 3.1 shows how different disturbances affect the different signals. In part (a) the compressor speed is step decreased, resulting in a significant change in

evaporator saturation temperature. Figure 3.1(b) shows how a step increase in the evaporator water pump speed results in an increase in the outlet temperature (and hence superheat) without affecting the inlet temperature, which is strictly a function of the evaporator pressure. Since a richer information stream can lead to better control, a controller that uses both of these signals without the additional complexity of a model-based approach is desirable; the cascaded control architecture features these characteristics.

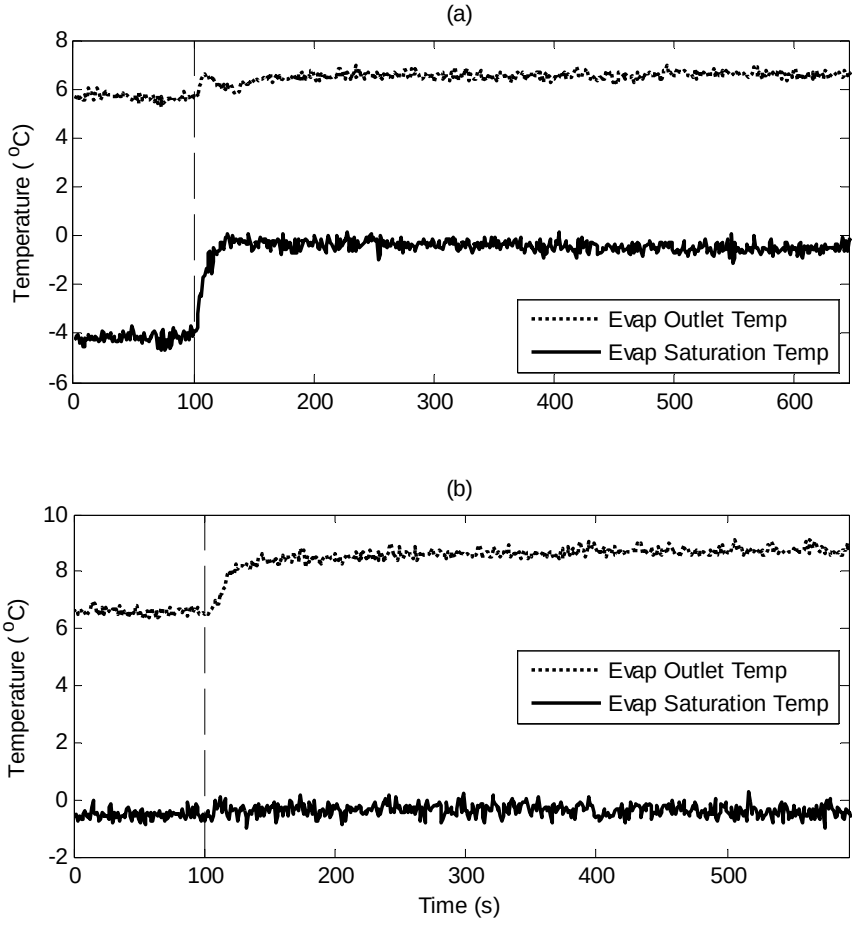


Figure 3.1: Evaporator inlet and outlet refrigerant temperatures in the presence of disturbances. (a) Step decrease in compressor speed. (b) Step increase in water flow (secondary fluid).

Cascaded Superheat Control

The cascaded controller combines two control loops: a high-gain proportional control loop that uses the valve position to regulate evaporator pressure to a setpoint, and an outer control loop that generates a pressure setpoint for the inner controller to meet. Refer to Figure 3.2 for a block diagram. The inner controller should have as high a gain as possible to ensure fast control action. The outer controller is generally a proportional-integral (PI) or proportional-integral-derivative (PID) controller, and is tuned to change slowly. In this way the inner loop handles "internal" disturbances that occur in the vapor compression system itself, e.g. compressor speed changes or evaporator shutdown/activation in a multiple evaporator system. The outer loop will change the inner setpoint to respond to slower changes, such as changes in the temperature of the zone being controlled. Successful implementation of this architecture requires a fast-acting expansion valve; since many electronic expansion valves generally have delays and slew rate limits, this is an important consideration in selecting components. The next section displays a set of simulations demonstrating the importance of actuator selection for this particular architecture.

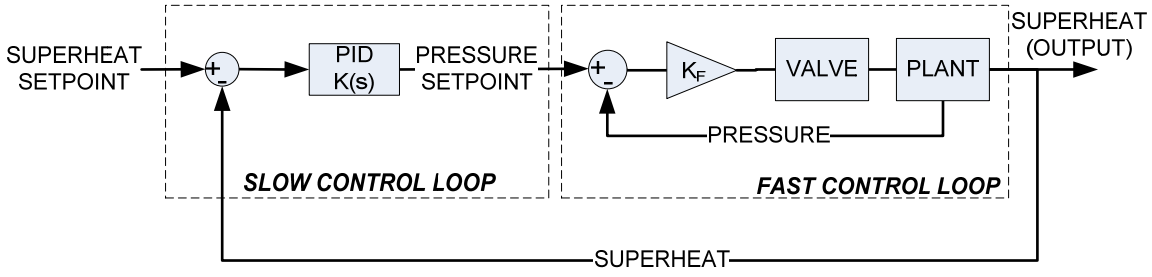


Figure 3.2: Cascaded control architecture.

Simulation of Actuator-Evaporator Dynamics

Standard commercially available electronic expansion valves (EEVs) use a stepper motor with a needle valve to meter refrigerant, which will have a maximum rate of change (slew rate) and a delay between the signal and the response of the motor. In order to examine the effects of these limitations on the response of the valve in a cascaded control loop, simulations were performed with a nonlinear evaporator and EEV model. A cascaded control loop was used to control superheat with EEV position. Slew rate limits and delay were added to the model as detailed later. While actuator saturation is a factor—valves cannot open past 100% or close past 0%—proper sizing of the valve prevents saturation from being a problem in system operations, and is therefore not explored here.

In general, the dynamics of vapor compression systems are highly nonlinear and complex; a complete discussion of first-principles modeling is outside the scope of this thesis, but reviews can be found in [93-96]. The plant model used in the following simulations is a nonlinear moving boundary model generated using techniques detailed in [97-99].

For the first set of simulations, shown in Figure 3.3, the effects of slew rate limits are examined during a step increase in the compressor speed. Superheat is controlled using the cascaded control architecture. The gains in the PI and proportional controllers were optimized for disturbance rejection using a gradient-descent method [100]. To establish the best case scenario, no slew rate limit was placed upon the valve during the optimization. For the next two simulations, slew rate limits of 12 and 6% per second

were imposed on the valve with the same controller. The simulations show that moderate slew rate limitations do not impact the response of the controller, but for a severe limitation of 6% per second, the closed loop system is rendered unstable. Experience indicates that commercially available EEVs can have slew rates from 5% to 20% per second, with delays up to one second.

For the second set of simulations, shown in Figure 3.4, the same cascaded controller is used on an EEV with an input/output delay between the controller and the actuator. This simulates a common effect in stepper-motor driven EEVs. Delays of 0.1, 0.5, and 1.0 seconds are used, along with the baseline case of no delay. The disturbance for this case is a step decrease in compressor speed. The delays have a stronger effect upon the success of the controller, although they do not render the plant closed loop unstable. For a delay of greater than 0.5 s, the systems exhibits nonlinear limit cycle behavior.

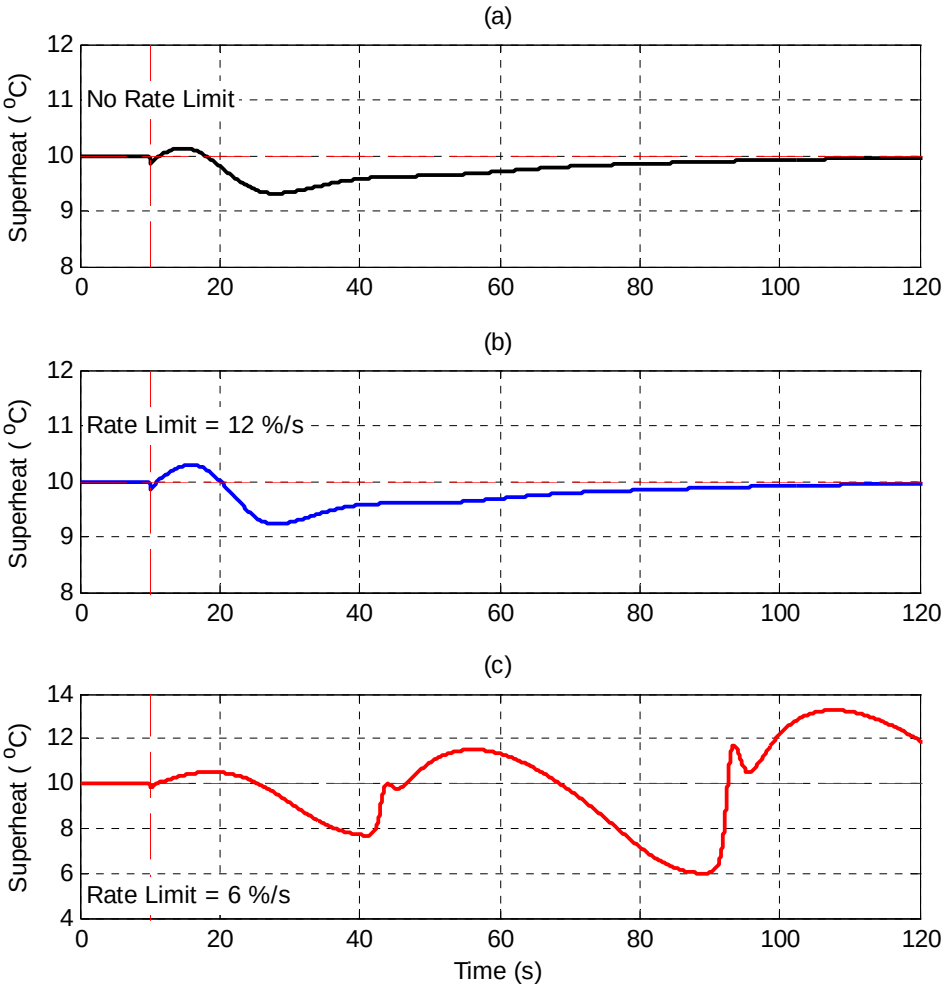


Figure 3.3: Disturbance rejections for actuators under rate limits. (a) no limit, (b) 12% per s, and (c) 6 % per s.

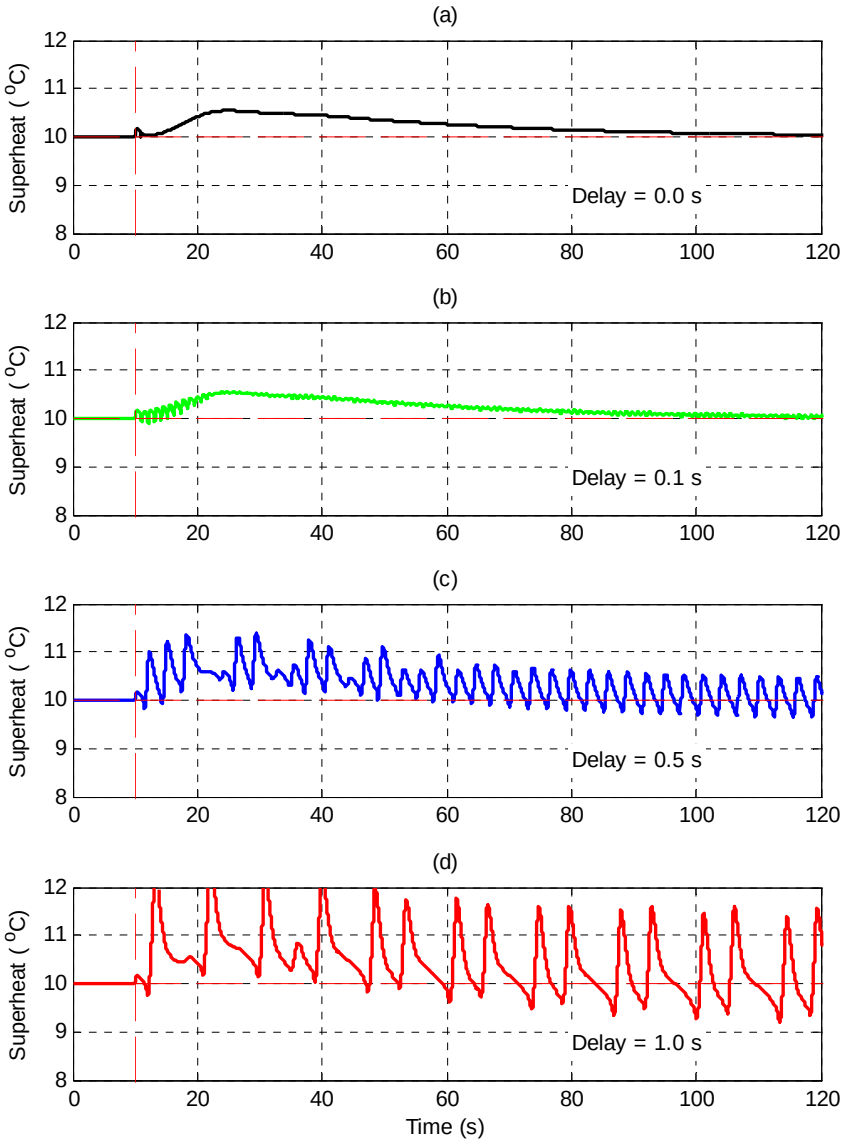


Figure 3.4: Disturbance rejection in the presence of I/O delays. (a) no delay, (b) 0.1 s delay, (c) 0.5 s delay, and (d) 1.0 s delay.

A final set of simulations was run with a system that had both a slew rate limit of 15% per second and an I/O delay of 0.5 seconds imposed upon the actuator. In order to gain a better grasp of the limitations imposed by these nonlinearities, the controller gains were re-optimized with the limitations in place. The disturbance rejection response of this system is superimposed on the response of an ideal system with instantaneous response and no delay, seen in Figure 3.5. Even though the limited actuator case is optimized, it still responds more slowly than the ideal case, suggesting that standard EEVs may have difficulty implementing the cascaded control architecture. The optimized gains for these two situations are shown in Table 3.1. Note in particular that the inner feedback gain K_F is significantly smaller for the limited case; the pressure regulating inner controller will not be as aggressive in seeking the setpoint, resulting in worse performance for the control architecture. The next section details a set of experiments carried out to evaluate the performance of different actuators under the cascaded control regime.

Table 3.1: Comparison of Controller Gains

Actuator	K_P	K_I	K_D	K_F
Ideal EEV	16.68	2.27	0.0004	7.16
Limited EEV	14.61	0.718	-1.39	1.45

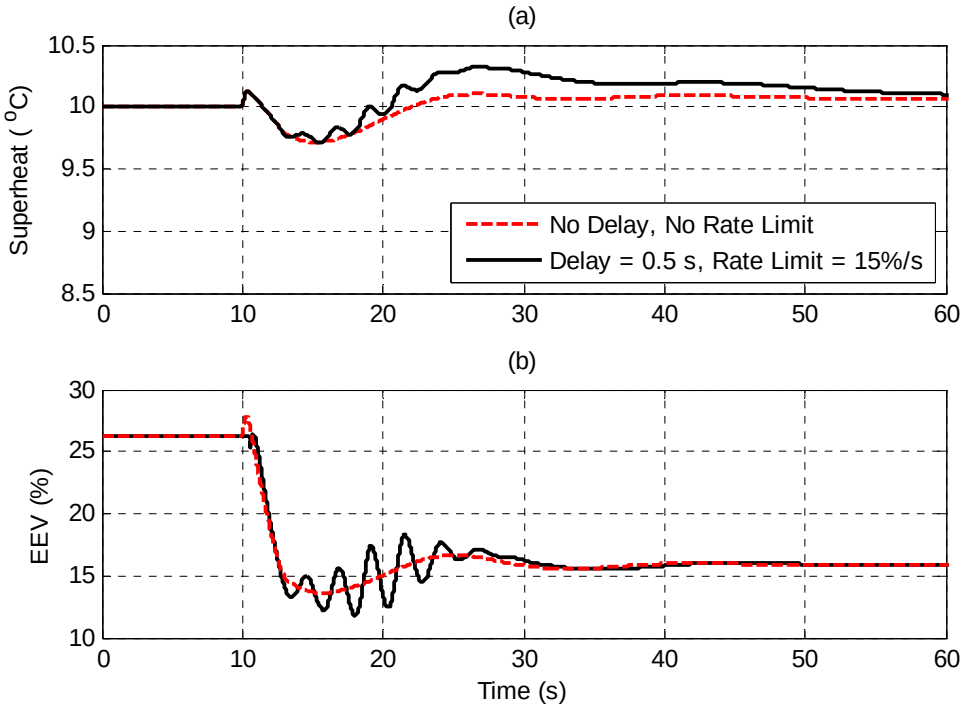


Figure 3.5: Optimized disturbance rejection for limited and ideal actuators. (a) superheat response, and (b) actuator effort

Experimental Evaluation

Testing Regimen

Most VCC cycles are subjected to a continuous stream of transient disturbances (i.e. opening the refrigerator door, fluctuating ambient temperatures, changes in thermostat settings, etc.). However, automotive air conditioning systems experience perhaps the most extreme of transient environments. Not only does the system experience the standard disturbances to cooling capacity demands, but changing vehicle speed results in transient disturbances of air flow rates. Most significantly, the compressor is generally driven by an accessory belt tied directly to the engine, so its

speed varies with the engine speed through both vehicle speed changes and gear changes.

The suitability of three different actuators for use in cascaded control will be evaluated using a standard automotive test cycle adapted from the FTP-SCO3 test protocol and designed to simulate the vehicle and engine speed during a short commute. The compressor speed profile is shown in Figure 3.6. To counteract the effects of these disturbances, the expansion valve is used to regulate evaporator superheat. These tests are conducted on the experimental test bed described in Chapter II.

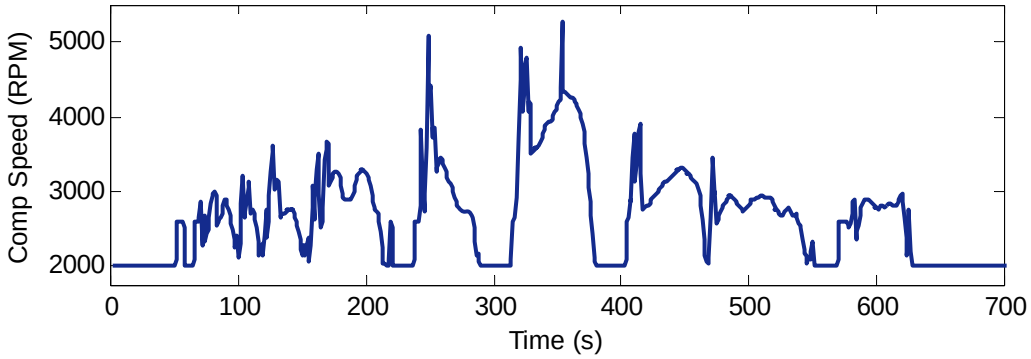


Figure 3.6: Compressor speed profile, severe automotive test

Actuators

Three different actuators will be used to evaluate the cascaded control architecture for superheat. The first is the standard Sporlan EEV described in Chapter II.

The second actuator is based upon micro-electrical-mechanical systems (MEMS) technology. Referred to as a Silicon Expansion Valve, or SEV, it is a proportional direct acting MEMS valve prototype manufactured by Microstaq, Inc. specifically as an

HVAC&R expansion device. The valve itself is thermally actuated silicon and accepts a pulse width modulating (PWM) signal and opens from 0% to 100% based upon the command signal, and thus from the operator's perspective behaves exactly like an EEV. However, the SEV design features few moving parts, an extremely small device size, and a response time measured in milliseconds rather than seconds. Additionally, since the valve is manufactured from a silicon wafer, there is no risk of fatigue failure, provided the chip is properly fabricated. A photo of the valve's internal components is shown in Figure 3.7; the silicon wafer is indicated in the photo.

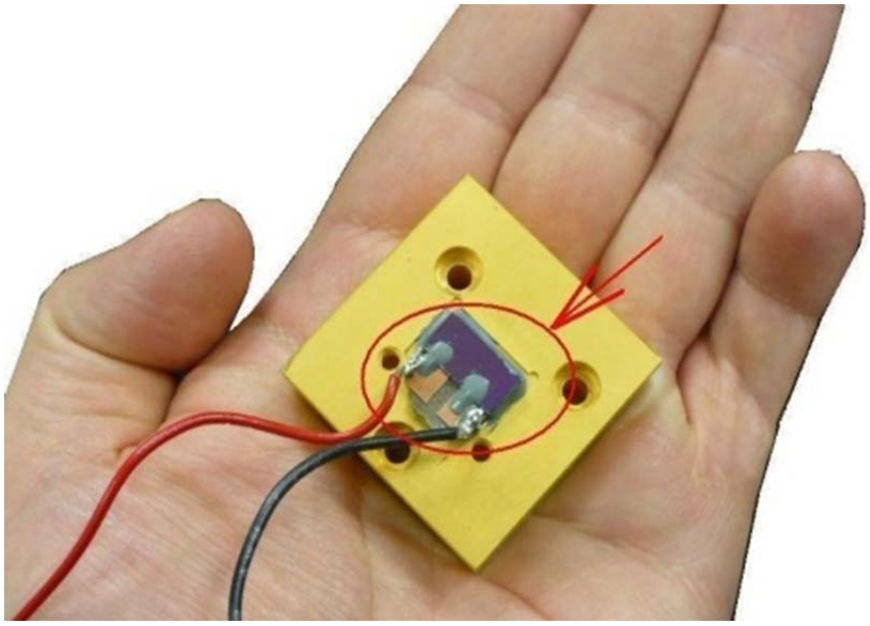


Figure 3.7: MEMs based Silicon Expansion Valve prototype (SEV)

The final actuator tested is a hybrid electro-mechanical expansion valve (HEV), first presented in [101]. This device features a mechanical expansion valve that

regulates the evaporator pressure with a mechanical diaphragm; the pressure setpoint is adjusted by varying the force applied to the top of the diaphragm, customarily by turning a set screw. The set screw is replaced with a linear stepper motor attached to the valve; the stepper motor's position is controlled by a computer signal. By using a PID controller to track superheat by changing the mechanical pressure setpoint, the HEV becomes an implementation of the cascaded control architecture. A schematic of the HEV arrangement is given in Figure 3.8.

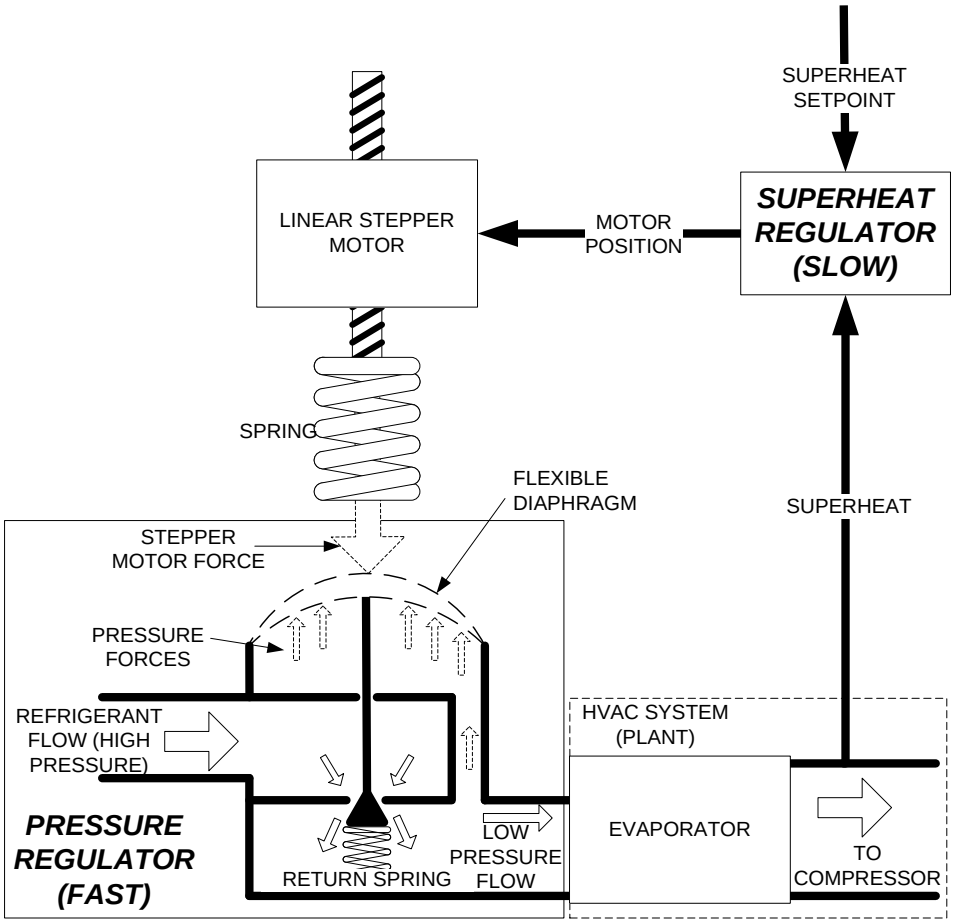


Figure 3.8. Hybrid expansion valve design.

All three actuators will be tested in the presence of the automotive-style compressor speed disturbances. The EEV and SEV will be set up with a purely electronic cascaded control architecture. See Figure 3.9. Table 3.2 shows the tuned controller gains for each of the control configurations. The EEV has a slew rate limit of 12.5% per second, a delay of approximately 1.0 s, and a rated capacity of 1.75 kW.

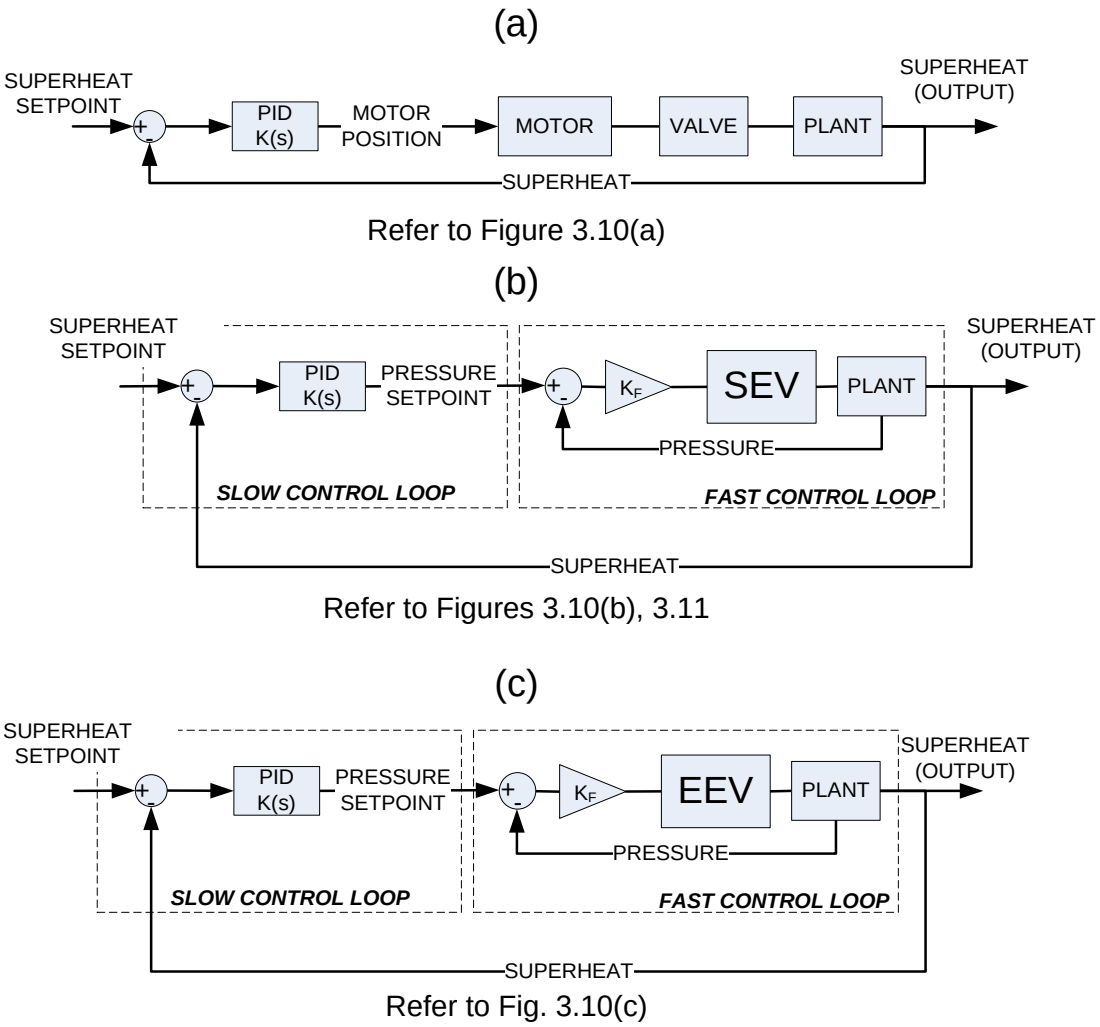


Figure 3.9: Control architectures

Table 3.2: Controller Gains

Control Loop	K_P	K_I	K_D	K_F
HEV (Figure 3.9a)	1.5010(a)	0.20	0.10	N/A
SEV/Cascaded (Figure 3.9b)	9.60	2.00	0.25	4.00
EEV/Cascaded (Figure 3.9c)	1.00	2.50	2.00	0.22

Evaluation Criteria

In order to quantify the performance of the controllers, the following characterizations are used. First, the root mean square error (e_{RMS}) gives a measure of the deviation from the setpoint is calculated thus:

$$e_{RMS} = \sqrt{\sum_{i=1}^{n} \left(sh_i - sh_{set} \right)^2} \Big/ \sqrt{n} \tag{3.1}$$

In this equation sh_i is the superheat at sample instant i, sh_{set} is the superheat setpoint, and n is the total number of sample points. The maximum absolute error (e_{MA}) is calculated to give the maximum deviation from setpoint over all sample points:

$$e_{MA} = \max_i \left| sh_i - sh_{set} \right| \tag{3.2}$$

Experimental Results

With the compressor speed profile given in Figure 3.6, the responses of the HEV with PID control, the SEV with cascaded control, and the EEV with cascaded control are shown in Figure 3.10. Results are quantified in Table 3.3.

Table 3.3: Drive Cycle Results Comparison

Actuator	Figure	e_{RMS}	e_{MA}
HEV	3.10(a)	0.41	1.44
SEV	3.10(b)	0.66	3.97
EEV	3.10(c)	1.45	9.04

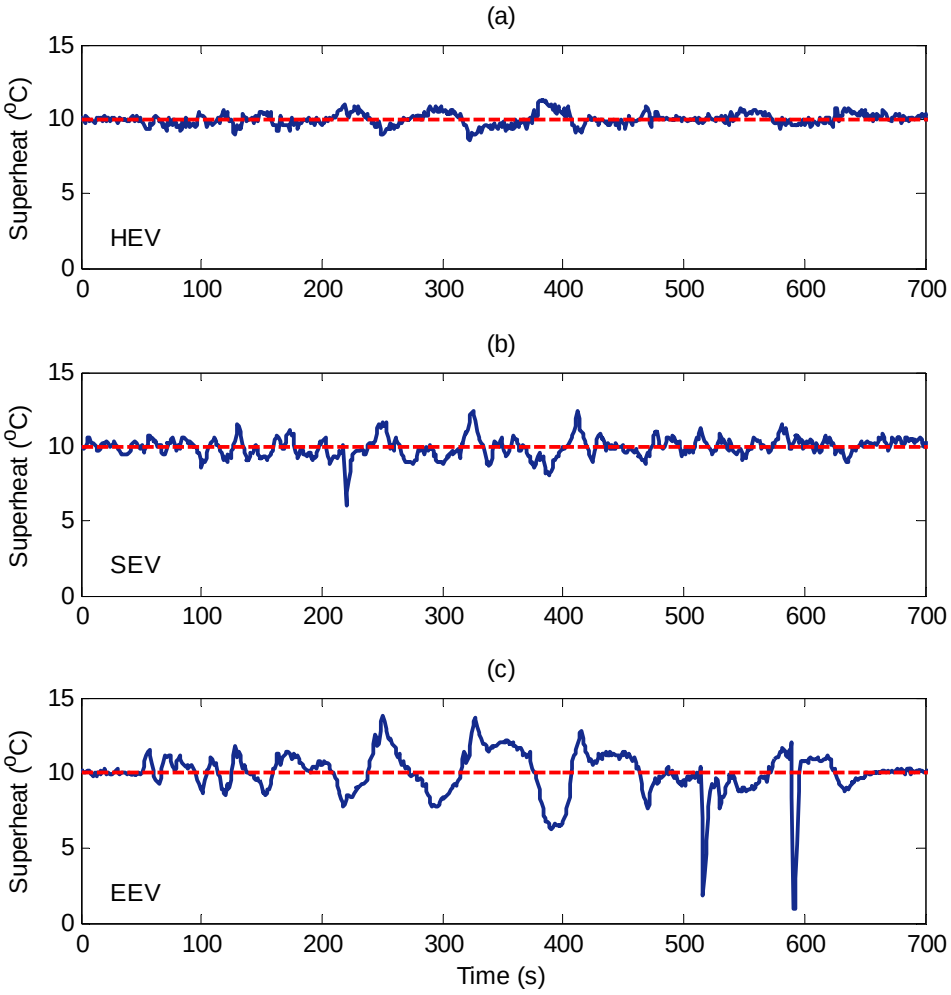


Figure 3.10: Disturbance rejection, automotive compressor profile for different actuators. (a) HEV—refer to Figure 3.9(a), (b) SEV with cascaded control—refer to Figure 3.9(b), (c) EEV with cascaded control-based emulation—refer to Figure 3.9(c).

These tests show that a cascaded control loop can successfully be implemented with either a mechanical hybrid system (the HEV) or a MEMS-based expansion valve. However, since EEVs are widely available and MEMS-based SEV's are a new technology, the question arises whether the EEV can be used in this new emulation architecture instead of the SEV. Unfortunately, the EEV-based emulation (Figure 3.10(c)) performs markedly worse than the SEV based emulation (Figure 3.10(b)). Indeed, the controller almost loses superheat completely in two different instances. As indicated by the earlier simulations, the limitations upon the EEV electronics and hardware prevent the EEV from responding fast enough to the internal disturbances to effectively serve as an HEV emulator.

To provide further evidence that these limitations are the cause of the problem, an additional test was run with these same limitations artificially imposed upon the SEV (Figure 3.11). When the EEV's actuator performance limitations are placed upon the SEV, it is unable to perform as needed. Given that the performance is worse than the EEV based emulation suggests that the rate limits and delays may be overestimated, but the principal result holds: actuator limitations can prevent achieving the best performance possible from a cascaded control loop architecture. The results suggest that for cascaded control the actuator must have the near-instantaneous response of a mechanical diaphragm (as in the HEV), or the high bandwidth performance of a MEMS-based actuator.

While the HEV does exhibit slightly better performance, the emulation approach with the SEV has several advantages. First, the valve embodiment is significantly

reduced in size; the bulky stepper motor and associated electronic hardware can be eliminated. Second, the SEV has no risk of diaphragm failure due to metal fatigue or motor failure; thus the increased control effort required over that of the HEV does not have a negative impact on using the SEV as a possible embodiment of the hybrid control approach. Third, the complete electronic configuration gives the flexibility to apply advanced control techniques (e.g. predictive control, nonlinear control, etc.) to this cascaded configuration, if desired, which is not possible using a mechanical feedback loop.

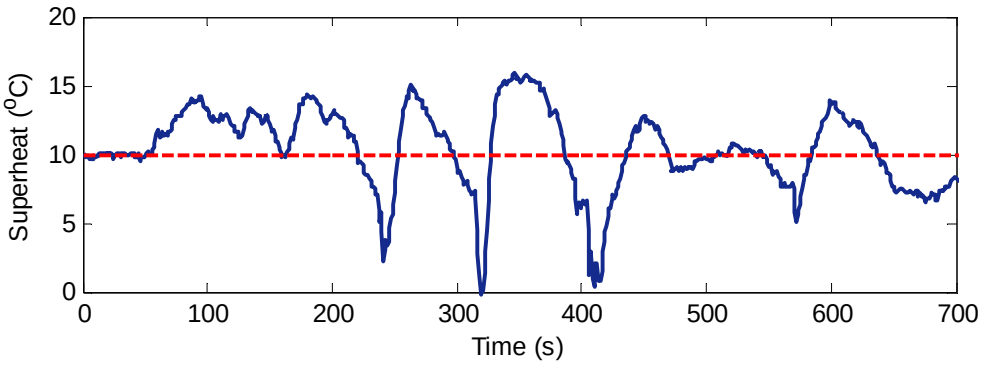


Figure 3.11: Superheat regulation with SEV when EEV's limitations are imposed. Compressor profile is the same as that of Figure 3.6.

Multiple Evaporator Systems

An increasingly frequent VCC configuration is the multiple evaporator system, sometimes referred to as a split system or variable refrigerant flow (VRF), where each room or zone being cooled has a separate evaporator [91]. Superheat control for these systems can be challenging due to dynamic cross-coupling between evaporators. The

cascaded control architecture successfully controls these systems as well, as is shown in the next two tests performed on the experimental system.

For the first test, shown in Figure 3.12, the compressor speed is increased and then decreased. Even though the evaporator dynamics are coupled, the cascaded controllers are able to successfully reject the disturbances to superheat, which has a setpoint of 8°C.

An important operating characteristic of VCC systems is pull down time, which is how long it takes the evaporators to reach an operating pressure. The faster that pull down is achieved, the faster the system is able to start cooling effectively; a fast pull down time means the compressor is consuming power for a shorter duration, and therefore implies a more energy efficient operating cycle. Due to the inner pressure control loop, cascaded controllers are particularly successful at achieving a fast pull down time. Figure 3.13 shows the pull down for the three-evaporator system (superheats and evaporator pressure). The controllers successfully bring the systems to their steady state operating conditions quickly, allowing the system to begin cooling rapidly.

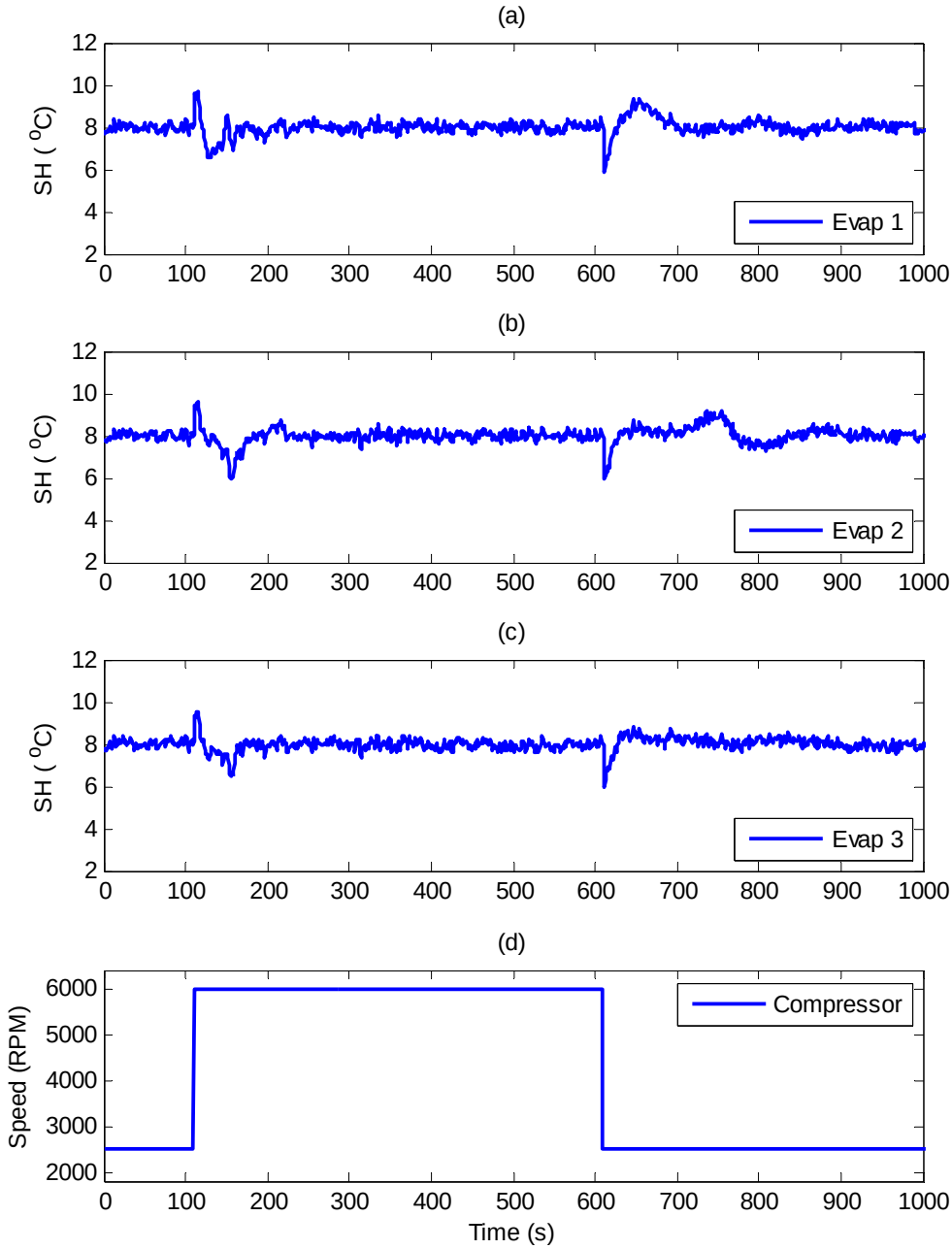


Figure 3.12: Step disturbance to three evaporator system (a) Evaporator 1 superheat, (b) Evaporator 2 superheat, (c) Evaporator 3 superheat, (d) compressor speed disturbance.

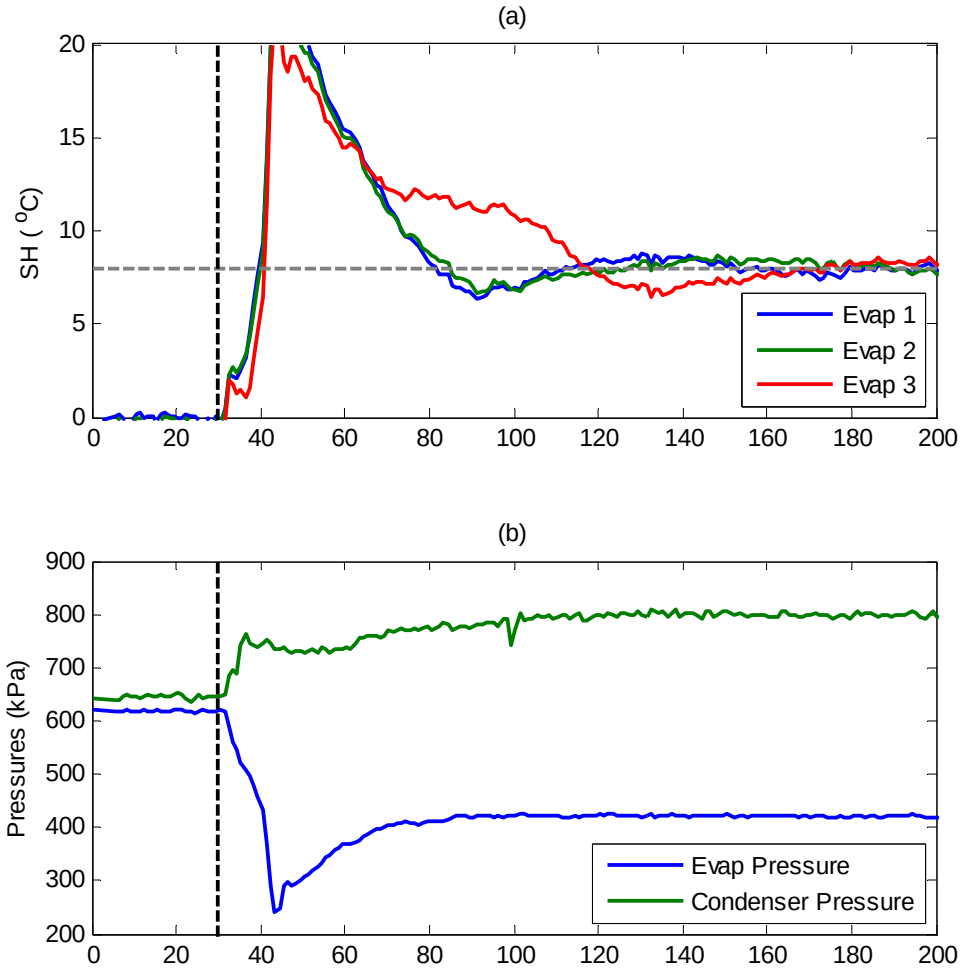


Figure 3.13: Pull down test results (a) Evaporator superheats (b) Evaporator and condenser pressures

Nonlinearity Compensation

An additional benefit to the cascaded approach to superheat control is that it also partially compensates for system nonlinearities without a priori knowledge of the system's characteristics or a mathematical model. This is a significant improvement, since EEVs in general are not a linear actuator, and the ability to compensate for system nonlinearities can lead to better performance over a wide range of operating conditions without techniques such as gain scheduling.

Inspection of refrigerant mass flow as a function of valve position for a typical EEV reveals a nonlinear relationship, as shown in Figure 3.14(a). Figure 3.14(b) shows superheat responses to EEV step changes for high refrigerant flow and low refrigerant flow conditions. Clearly, the response is much different for the EEV at high and low flow conditions. This is caused by the same step change causing different changes in flow depending on the valve position. This nonlinearity in the EEV means that adequate superheat control is much more difficult to achieve for this device, and illustrates why a gain scheduling approach based upon the different flow conditions is frequently used. Alternatively, the controller can be "de-tuned," sacrificing performance for stability over all operating regions due to smaller control gains; however, this risks losing superheat during system transients. Using a static inversion mapping between command signal and valve position [102], refrigerant flow was linearized, similar to the work presented in [103]. This resulted in a much smaller difference in step response for the different conditions, although differences in dynamic response are still evident. See Figure 3.15.

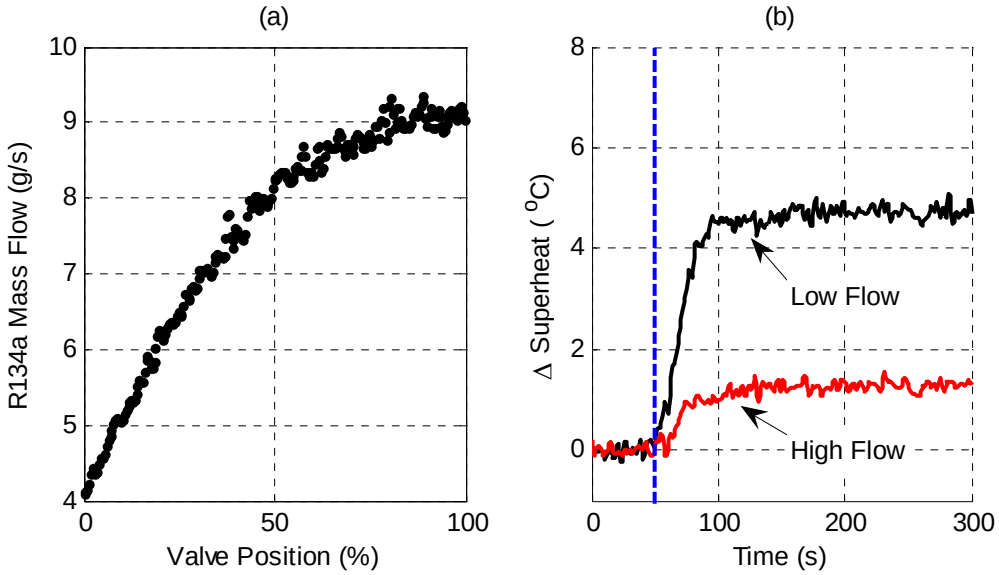


Figure 3.14: Nonlinearity of response for EEV. (a) Mass flow as function of valve position (static valve map) and (b) Superheat response to step change in valve position for high and low flows.

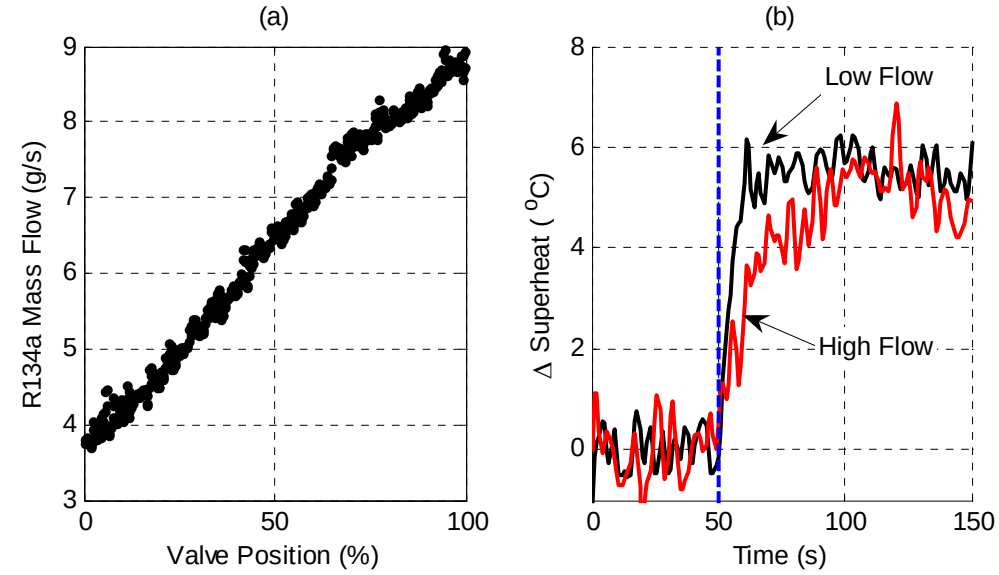


Figure 3.15: Linearized valve response. (a) Mass flow as function of valve position (static valve map) and (b) Superheat response to step change in valve position for high and low flows. Note the difference in speed of response, even though steady state response is very similar.

This result suggests that if the refrigerant flow is a linear function of valve command, the resulting superheat response will be substantially linearized. This may render the use of gain scheduling unnecessary, since robust control techniques can then be employed without unacceptable decreases in control performance. However, since EEVs will not in general be linear in flow characteristics, and since mass flow measurements may not be available for development of static inversion maps, a method for linearizing the flow (and hence the superheat response) is needed that only requires more basic measurements such as pressure and temperatures. The cascaded control approach proposed herein meets these requirements.

Modeling as a Nonlinear Feedback System

To analyze the nonlinearity compensation of the cascaded control architecture, the system is modeled as shown in Figure 3.16. As before, the inner, "fast" loop uses a proportional controller with gain K_F that seeks to regulate the evaporator pressure to a setpoint (P_{SET}) generated by an outer, "slow" controller *C(s)*; this pressure setpoint is chosen by the controller *C(s)* to regulate evaporator superheat to a user-defined setpoint. However, the relationship between valve position u and mass flow m is treated as a nonlinear gain function $K_M(v)$, and is a characteristic of the actuator used. The transfer functions *G(s)* and *H(s)* are the dynamic relationships from mass flow to superheat and pressure, respectively.

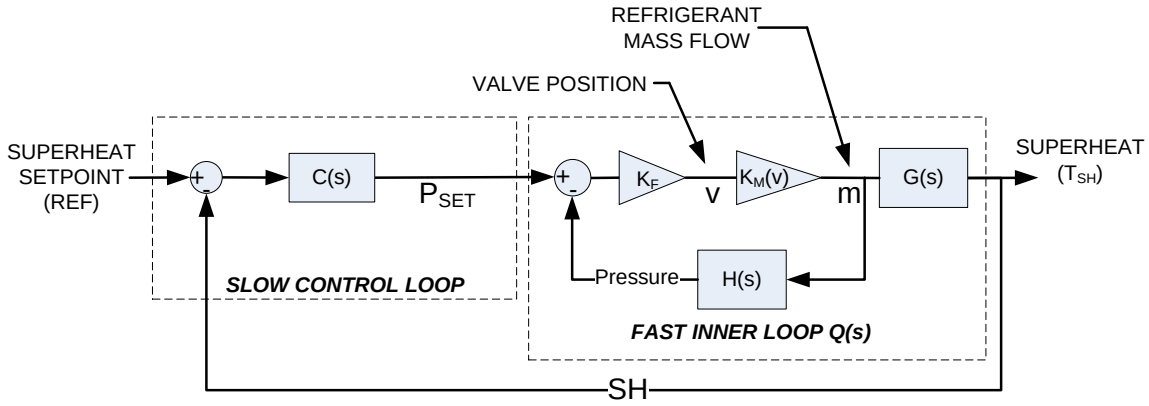


Figure 3.16: Cascaded control loops with transfer functions and nonlinear gains.

The fast inner loop block diagram can be reduced to a transfer function *Q(s)*:

$$Q(s) = \frac{sh}{p_{set}} = \frac{K_F K_M(v) G(s)}{1 + K_F K_M(v) H(s)} \tag{3.3}$$

Assuming stability allows invocation of the Final Value Theorem [104], which gives a value for the steady state gain of *Q*:

$$Q(0) = \frac{sh_{final}}{p_{set}} = \frac{K_F K_M(v) G(0)}{1 + K_F K_M(v) H(0)} = \frac{K(v) G(0)}{1 + K(v) H(0)} \tag{3.4}$$

Thus, if the steady state gains of *G* and *H* vary in the same direction as mass flow changes, e.g., they both decrease with increasing mass flows, then the variation of *Q(0)* as mass flow changes will be minimized as *K(v)* becomes larger. Experimental evaluation of the same EEV used earlier gave the following result, shown in Figure 3.17.

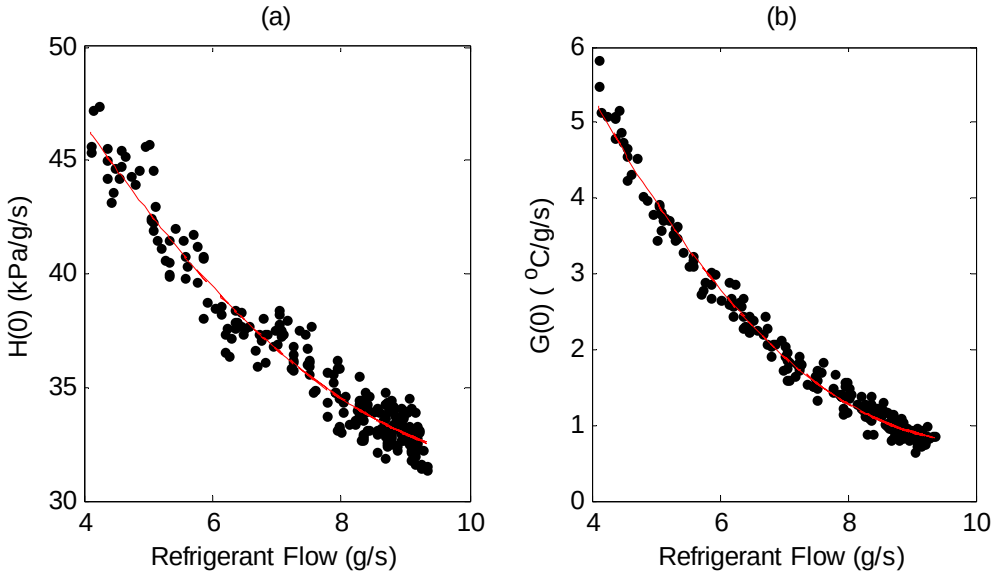


Figure 3.17: Steady state values of transfer functions. (a) H(s)—mass flow to pressure and (b) G(s)—mass flow to superheat. Both are shown as functions of refrigerant flow.

Clearly, the steady state responses of both *G(s)* and *H(s)* decrease with increasing mass flow. From Figure 3.17, a set of least squares curve fits was generated to give the relationships between valve position and mass flow, and mass flow to pressure and superheat. Using these functions in Equation 3.4 gives the curves seen in Figure 3.18. This figure shows the steady state gains for *Q(s)—*p_{set} to *sh—* and $K_M(v)\cdot G(s)$*—v* to *sh—* from their respective inputs. Clearly, there is a much more significant difference in the plant gain for the more traditional EEV-to-superheat control paradigm, than for the proposed p_{set} to superheat cascaded control architecture. Note that both the EEV and p_{set} values were scaled for this analysis so that the maximum for each value is 1; this allows an even comparison for the two paradigms.

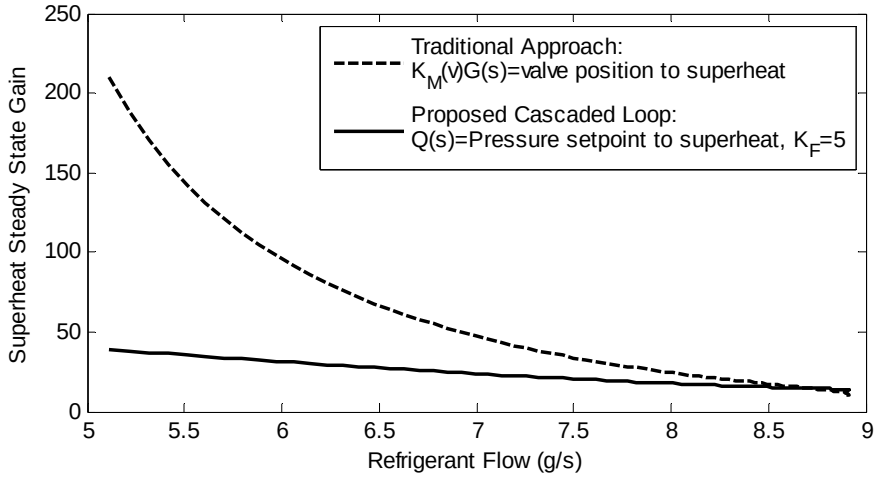


Figure 3.18: Steady-state Superheat Gain as a function of flow for EEV and Cascaded Loop

A final comparison is given in Figure 3.19. Both cases show improved linearity over the original nonlinear EEV's step response; the HEV is much more linear in its response, as is expected due to its higher internal gain.

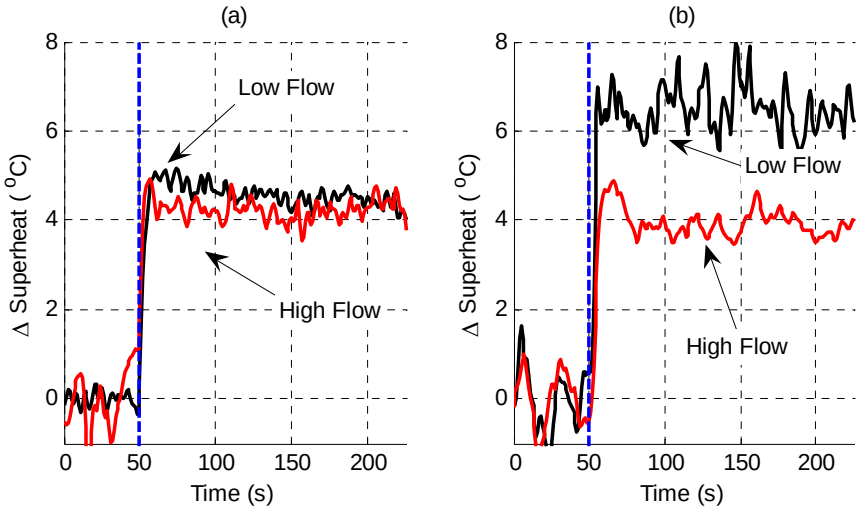


Figure 3.19. Superheat step responses at low and high flows. (a) HEV and (b) cascaded control with EEV.

This chapter presented a local control architecture for use with individual components in an HVAC&R network; namely, the expansion valves that control evaporator superheat. This architecture has been shown to provide improved control of superheat under various conditions, and is implementable in a variety of ways, including in a multiple evaporator system. Further details can be found in [101, 105, 106]. An additional benefit is the significant linearization of evaporator dynamics that can be implemented without a priori knowledge of the system model or the valve's flow characteristics; this can make implementation of effective superheat control in a large networked HVAC system much easier as well. More discussion of this linearization can be found in [107, 108].

CHAPTER IV

HIERARCHICAL MPC OF MULTIPLE EVAPORATOR SYSTEMS

The previous chapter explored a technique for controller design at the component level; with the implementation of good component control, higher level hierarchical and system-level controls become easier to achieve. This chapter looks at the system level and presents a control architecture for treating a multiple evaporator vapor compression cycle (ME-VCC) as a network of several logically divided subsystems. This architecture achieves system-level improvements in energy efficiency without the need for a centralized controller and the corresponding system model, which would be highly complex and difficult to obtain for systems with many evaporators spread over a large physical area. The control scheme is a two-level hierarchy that uses model predictive control (MPC) for the supervisory controller and the evaporator controllers; the pressure control is achieved with proportional-integral control. This is a refinement and improvement on the control scheme originally proposed in [109].

The chapter is organized as follows. First a general method for describing the system-level optimum is developed. Then the gains from the various actuators to the system outputs are explored, with the goal of dividing the ME-VCC system into a network of subsystems. A suite of local controllers is developed with this knowledge; these controllers take advantage of characteristics of the components and the model predictive control (MPC) algorithm to provide superior system control with reduced actuator effort. The control architecture is then applied to the experimental system; the

design process is covered in detail. Finally, the experimental results are presented and discussed.

System Optimum and the Supervisory Controller

This control architecture has a hierarchical structure wherein a supervisory controller generates setpoints for lower level controllers to meet. The lower level controllers in turn control the system actuators at the component level.

The supervisory controller is an MPC controller that chooses a control profile over N_u time steps that minimizes a cost function based upon the dynamics of the controlled plant over N_y time steps. This control profile consists of setpoints for the lower level controllers. A generalized version of the cost function is presented in this section; a specific example of how the cost function can be developed for a particular system is presented in the Controller Design section. Note that this cost function defines the system-level optimum for the purposes of this work; that is, the system is said to be at the optimal condition when the cost function is minimized.

The dynamics of the cooling zone temperature are much slower than the dynamics of the vapor compression system. In common experience, a VCC system can achieve reasonable setpoints (pulldown) on the order of 1-2 minutes if the controllers are well tuned, while cooling a room by several degrees can take 10-20 minutes. This time scale separation allows separation of the control functions, and ensures that the proposed hierarchy of controllers is a feasible design choice.

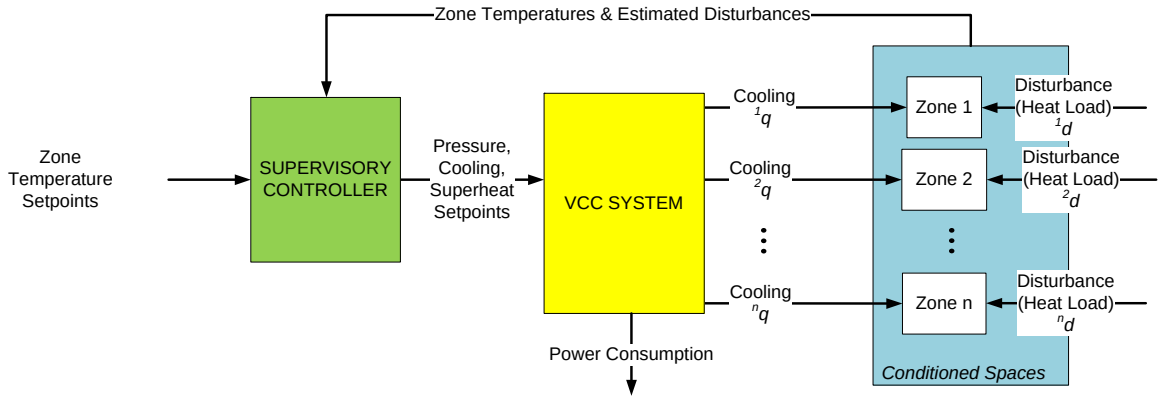


Figure 4.1: Supervisory controller and relationship to multi-evap cooling system with *n* zones.

The supervisory controller treats a VCC system with *n* evaporators as a MIMO plant whose inputs are evaporator pressure, superheat, and cooling rate setpoints. See Figure 4.1. The outputs are the zone temperatures, which are set (e.g., with a thermostat) by the user or occupant of the zone, and the system's power consumption, which does not track a setpoint, but plays an important role in the supervisor's calculation, as will be seen later. The i^{th} zone can be represented as a discrete-time dynamic system thus:

$$ {}^{i}T_{k+1} = {}^{i}f({}^{i}T_k, {}^{i}q_k, {}^{i}d_k) \tag{4.1} $$

If ${}^{i}T_{set}$ is the user-defined setpoint for zone temperature *i*, the error at time *k* is then:

$$ {}^{i}e_k = {}^{i}T_{set} - {}^{i}T_k \tag{4.2} $$

For a system with *n* zones, an error vector can therefore be constructed for time step *k*:

$$ e_k = \left[{}^{1}e_k \;\middle|\; \cdots \;\middle|\; {}^{n}e_k \right]^T \tag{4.3} $$

Since the supervisor must balance zone temperature setpoint tracking with energy consumption, a means of expressing the instantaneous system operating

efficiency at sample time *k* must also be developed. A common way of expressing the efficiency of VCC cycles is the Coefficient of Performance (COP), defined as the ratio of cooling performed by the system to the total work input required by the system [110]. Since COP increases with increasing efficiency, and the controller is seeking to minimize a cost function, the inverse COP (ICOP), designated β^{-1}, is used:

$$\beta^{-1}{}_k = \frac{\dot{w}_{comp}({}^{\min}p_k, {}^1sh_k, \cdots, {}^nsh_k, \sum_{i=1}^{n} {}^iq_k) + \sum_{i=1}^{n} \dot{w}_{fan}({}^ip_k, {}^ish, {}^iq_k)}{\sum_{i=1}^{n} {}^iq_k} \tag{4.4}$$

The relationships for power consumption by the compressor ($\dot{w}_{comp}$) and the evaporator fans ($\dot{w}_{fan}$) are polynomial functions of evaporator pressures, superheats, and cooling rates and are developed experimentally or from manufacturer data. The compressor power term is dependent on the evaporator superheats, the total system cooling, and the minimum evaporator pressure, ${}^{\min}p_k$, which corresponds to the suction pressure at the compressor. (Refer to the vapor compression cycle primer in Chapter I). By the merits of its construction, the ICOP function can be easily modified if the system components are changed. For example, each evaporator can have a different fan, although a different polynomial function $\dot{w}_{fan}$ is required for each. This gives a degree of flexibility and modularity to the supervisory controller.

The last ingredients for the cost function are vectors for the controller inputs and changes in inputs are presented:

$$u_k = \left[{}^1p_k \quad {}^1q_k \quad {}^1sh_k \;\middle|\; \cdots \;\middle|\; {}^np_k \quad {}^nq_k \quad {}^nsh_k \right]^T \tag{4.5}$$

$$\Delta u_k = u_k - u_{k-1} \tag{4.6}$$

$$\vec{u}_p = \left[u_k \mid \cdots \mid u_{k+Nu-1} \right]^T \tag{4.7}$$

Thus, at time *k*, the supervisor controller chooses the profile $\vec{u}_p$ that minimizes:

$$J_k = \sum_{j=k+1}^{k+Ny} \left(e_j{}^T \Lambda_e e_j + \Lambda_\beta \beta_j \right) + \sum_{j=k}^{k+Nu-1} \left(\Delta u_j{}^T \Lambda_\Delta \Delta u_j \right) \tag{4.8}$$

The next section examines the dynamics of a multiple evaporator system in order to develop a local set of control architectures to track the setpoints generated by the supervisory controller.

Local Control Architecture and Dynamic Analysis

The preceding section detailed the development of a supervisory controller that specified the cooling rate, superheat, and pressure for each evaporator in a multiple evaporator system. This section will use dynamic analysis tools and reasoning based upon the physical construction of the system to motivate the local architecture design decisions.

Multiple evaporator-based systems are by nature spatially distributed, since the evaporators are operating in physically separate cooling zones and the compressor and condenser are often outside or in an uncooled space. The independent system variables to be controlled are evaporator cooling, evaporator superheat (a measure of effective use of the heat exchanger), and the evaporator pressures. Since each evaporator has its own expansion valve, fan or pump, and optional discharge valve, using these actuators to control the evaporator's operating condition is a logical choice. Additionally, the

compressor can be used to control the suction pressure, which will be the same pressure as any evaporator without a discharge valve.

Dynamic Analysis

To ensure that the proposed architecture is generalizable, the dynamics of a variety of multi-evaporator systems were analyzed. Using a finite control volume approach [96], a simulation model was developed for a multi-evaporator air conditioning system with two evaporators. The model was linearized [111] about high and low capacity operating conditions to obtain a suite of single-input, single-output (SISO) models. Additionally, a set of experimentally identified linear models was created of an experimental multi-evaporator water chiller using standard system identification (ID) techniques [112]. The details of the system identification can be found in [109]. Despite the differences among these systems, their normalized frequency responses exhibit common trends.

Expansion valves are traditionally used solely to regulate superheat. Figure 4.2 shows, however, that the dynamic response of refrigerant side cooling to EEV changes is nearly as strong and notably faster, since changes in EEV opening alters refrigerant flow rate virtually immediately.

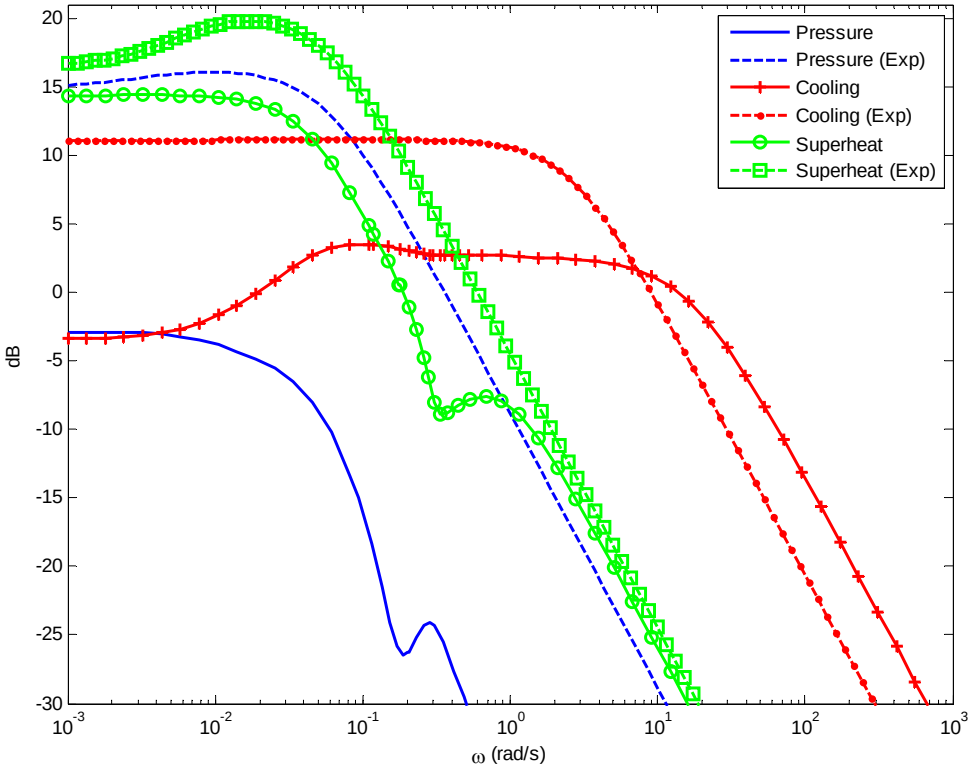


Figure 4.2: EEV position to Pressure, Cooling, and Superheat for experimentally derived and simulated systems.

The fan (Figure 4.3) also has a strong response to superheat. Given that these two actuators (EEV and fan) are spatially co-located for each evaporator, coordinating their actions to jointly regulate superheat and cooling is a logical choice both spatially and dynamically, and can be repeated for each cooling zone. Using an MPC-based controller here will allow system constraints to be respected; in particular, superheat can be restricted to a band and allowed to float within this band. This control design is discussed in more detail in the following section.

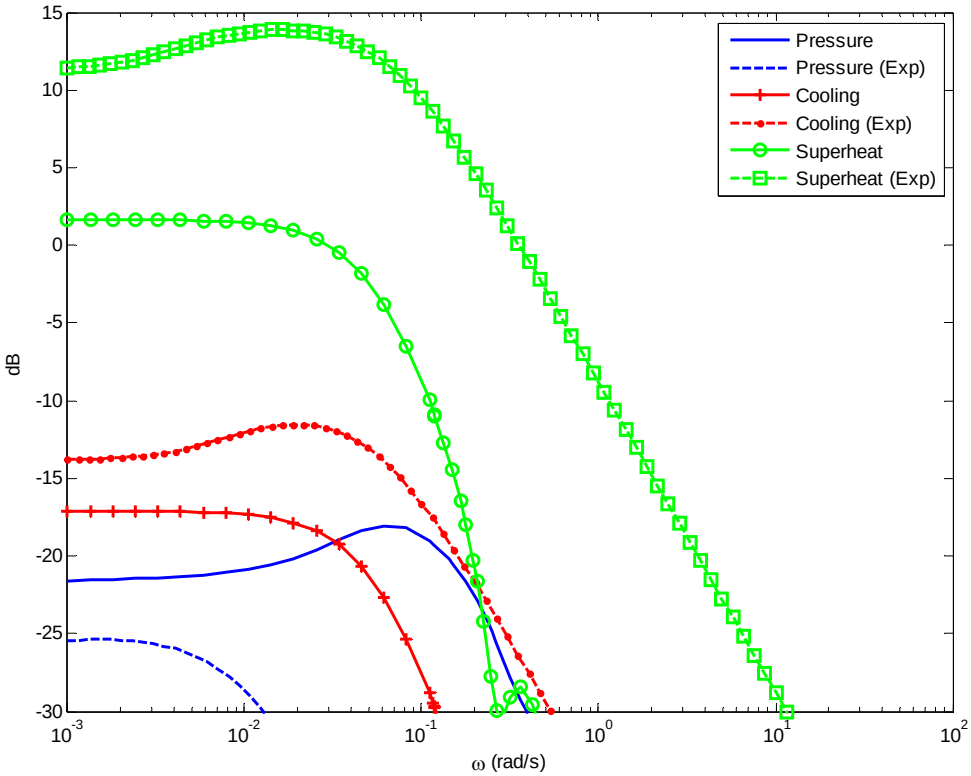


Figure 4.3: Fan position to Pressure, Cooling, and Superheat for experimentally derived and simulated systems.

The compressor has a consistently strong effect on evaporator pressure (Figure 4.4), motivating this particular input/output pairing. Although compressor speed also affects superheat, the compressor is generally located far from the evaporators in a VRF system. If the algorithm controlling the EEV and fan is either slow acting or allows deviations from setpoint, which the MPC controller will do, the effect of the compressor on superheat will not cause the controllers to fight each other, essentially decoupling of the effect of compressor action on the evaporator superheat.

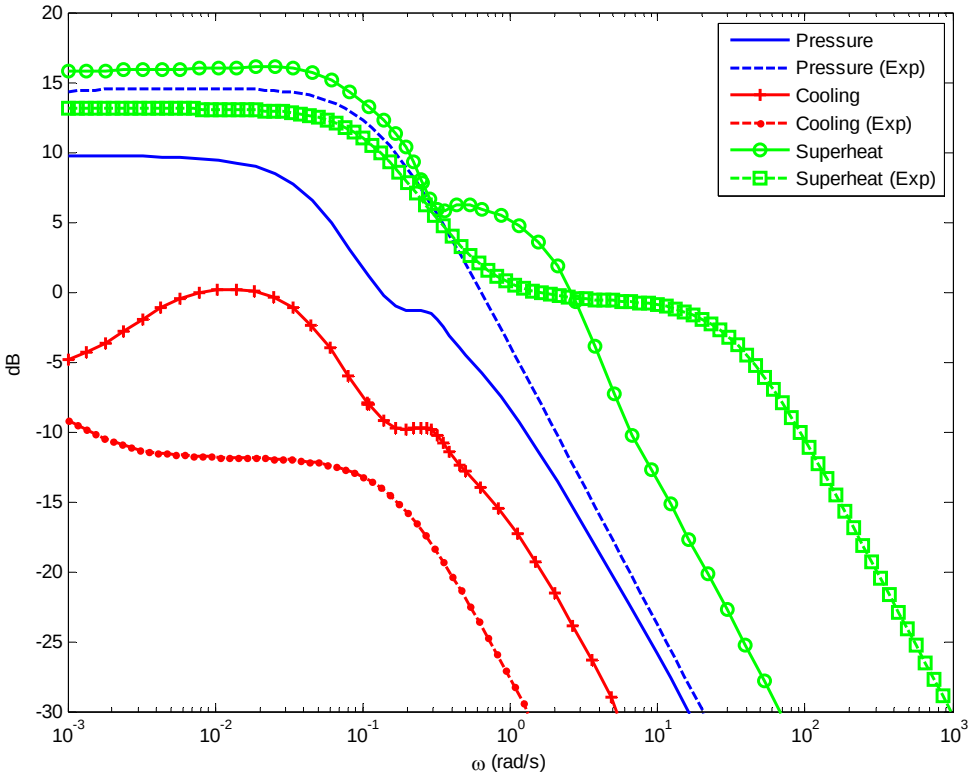


Figure 4.4: Compressor speed to Pressure, Cooling, and Superheat for experimentally derived and simulated systems.

Controller Architecture

Figure 4.5 shows the control architecture proposed in this paper for a system with *n* evaporators like that in Chapter I. The actuators are electronic expansion valves (EEVs) and variable speed fans (FAN) for each evaporator, as well as a variable speed compressor (RPM). Additionally, any evaporator that needs to operate at a higher pressure (i.e., evaporate at a higher temperature) will have an electronic discharge valve (DV). Measurements will be taken of the zone temperatures, the refrigerant temperatures at the inlet and outlet of each evaporator, the evaporator pressures, and the refrigerant flow into each evaporator. Although refrigerant flow is considered to be an expensive measurement, virtual sensors can successfully be used to estimate this flow [113].

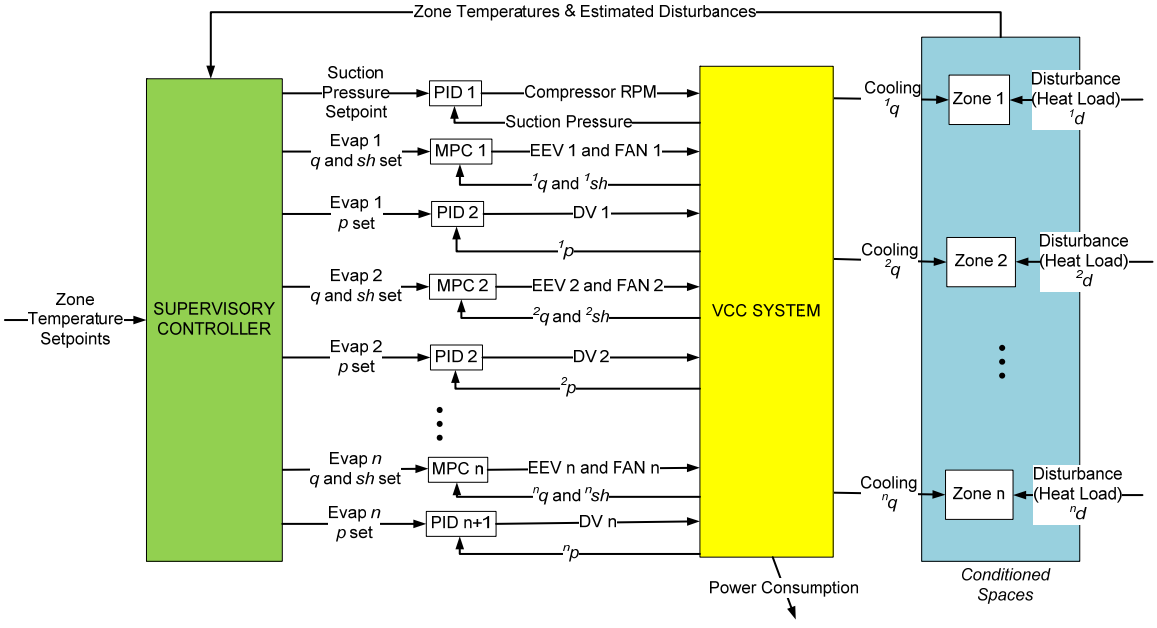


Figure 4.5: Control architecture

At the local level, the EEV and fan for each evaporator will be coupled with an MPC based controller to track an evaporator cooling rate setpoint. This controller will also regulate superheat so that it stays within a constrained band, e.g., between 6 and 12 °C. This approach is very different from the way that evaporators are generally controlled, in that the controller does not seek explicitly to track a superheat setpoint, and is only possible due to exploitation of the capabilities of MPC. Strict regulation of superheat to a setpoint is unnecessary, as long as superheat stays above a minimum value to protect the compressor. However, excessive superheat is undesirable, since a high superheat implies that a great deal of heat transfer is occurring when the refrigerant is in a gaseous state, which is much less energy efficient than the heat transfer that occurs during refrigerant evaporation. If the superheat band is set correctly, the controller and evaporator will deliver a steady amount of cooling and operate in a safe yet efficient manner.

The remaining actuators are used to control the evaporator pressures. The compressor speed is set by a proportional-integral-derivative (PID) controller to regulate the suction pressure. Separate PID controllers use the discharge valves to regulate the pressure in the evaporators as appropriate.

Since the controller for each evaporator is separate, evaporators can be added or removed, or equipment changed, without affecting the other evaporators. This adds an element of modularity to the control architecture that is highly desirable but not possible with completely centralized control. The proposed local control architecture thus gives each zone its own independent control package, with the compressor responding to the

cooling demands placed on the system as a whole. However, another level of control is needed to allocate cooling capacity to the different evaporators in order to regulate the zone temperatures. Thus the architecture also includes the supervisory controller detailed earlier to regulate the cooling zone temperatures and the energy efficiency of the entire system.

Controller Design for an Experimental Testbed

As detailed in Chapter II, the experimental system features three 1.5 kW capacity evaporators, each cooling a different 50 liter water tank that serves as a zone. There is a single condenser, with a variable speed compressor with a maximum cooling capacity of 5.58 kW. The reader should note that this system will mimic a system where the refrigerant evaporation directly cools the air in the room or cabin, and not a system where chilled water is used to cool the air in the various cooling zones. The other actuators are electronic expansion valves (EEVs) at the inlet of each evaporator and variable flow water pumps (WPs) to regulate water flow through the evaporators (analogous to variable speed fans in an air-based system). High and low side pressure measurements are available, as well as refrigerant mass flow, water temperature measurements at the inlet and outlet of each heat exchanger, and refrigerant temperature measurements at each point of the thermodynamic cycle. The system does not have discharge valves at the outlet of the evaporators, so only the suction pressure will be regulated. Also, the supervisory controller will only calculate one superheat setpoint for

all three evaporators. The next sections will detail how the generic architecture presented earlier is applied to this experimental testbed.

Supervisory Controller

The zone temperature dynamics are modeled with a first order lumped capacitance discrete time model with an input disturbance; in this model the output is the cooling zone temperature, the input is the evaporator cooling rate, and the disturbance is heat transfer into the cooling zone from parasitic heat losses or gains. For the i^{th} zone, which is cooled by the i^{th} evaporator, the model is defined as:

$$ {}^{i}T_{k+1} = {}^{i}T_{k} - \frac{t_s}{mc}\,{}^{i}u_{k} + \frac{t_s}{mc}\,{}^{i}d_{k} \tag{4.9} $$

In this equation, T_k is the cooling zone temperature at sampling instant k, t_s is the sample period, m is the mass of the water being cooled (50 kg), and c is the specific heat of water (4.186 kJ/kg/K). The input u_k is the evaporator cooling in kW, and the exogenous disturbance d_k is generated by an immersion heater; an observer [104] is used to estimate the size of the disturbance in kW, and this is fed to the supervisory controller. The observer is described by the following dynamical system:

$$ \begin{aligned} {}^{i}\dot{\tilde{x}} &= \left[A_{OB} - K_{OB}C_{OB}\right]{}^{i}\tilde{x} + \begin{bmatrix} K_{OB} & B_{OB} \end{bmatrix}\begin{bmatrix} {}^{i}T \\ {}^{i}q_k \end{bmatrix} \\ {}^{i}d_{est} &= \left[C_{OB}\right]\tilde{x} \\ A_{OB} &= \begin{bmatrix} 0 & 0.0048 \\ 0 & 0 \end{bmatrix}, \quad B_{OB} = \begin{bmatrix} -0.0048 \\ 0 \end{bmatrix} \\ C_{OB} &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad K_{OB} = \begin{bmatrix} 1 \\ 0.25 \end{bmatrix} \end{aligned} \tag{4.10} $$

In this equation ${}^{i}\tilde{x}$ is the state of the i^{th} zone's estimator and ${}^{i}d_{est}$ is the estimate of the disturbance. Since all three zones are identical, and use the same observer gains, all three can use the same estimator.

For the supervisory MPC controller, a set of matrices are developed by recursive substitution of the dynamic state matrices [5, 114], resulting in the following predicted temperature profile ${}^{i}\vec{T}_p$ that corresponds with a control profile ${}^{i}\vec{u}_p$:

$$ {}^{i}\vec{T}_p = {}^{i}F\,{}^{i}T_k + {}^{i}H\,{}^{i}\vec{u}_p + {}^{i}K\,{}^{i}d_k \tag{4.11} $$

With the following definitions for prediction matrices:

$$ {}^{i}F = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_{Ny\times 1}, \quad {}^{i}K = \frac{t_s}{mc}\begin{bmatrix} 1 & & 0 \\ \vdots & \ddots & \\ 1 & \cdots & 1 \end{bmatrix}_{Ny\times Ny} $$

$$ {}^{i}H = \frac{-t_s}{mc}\begin{bmatrix} 1 & & 0 \\ & \ddots & \\ & & 1 \\ \vdots & & 2 \\ & & \vdots \\ 1 & \cdots & (Ny - Nu + 1) \end{bmatrix}_{Ny\times Nu} \tag{4.12} $$

Note that although the cooling zones have been modeled with simple lumped capacitance assumptions, the proposed approach is extendable to more complex dynamic models of the cooled spaces, if necessary. For example, higher order models that better describe the particular space being cooled can be used. Disturbances to the room temperature such as outside heat load or parasitic heat losses between cooling zones can also be modeled to better reflect the system, although in this demonstration they are treated as an exogenous input.

The ICOP formula for the experimental system was developed using steady-state data. To obtain this data, the system was slowly walked through its operating range as defined by the compressor speed and a superheat setpoint fed to the superheat controllers. This was repeated at different water flow rates through the evaporators. See Figure 4.6.

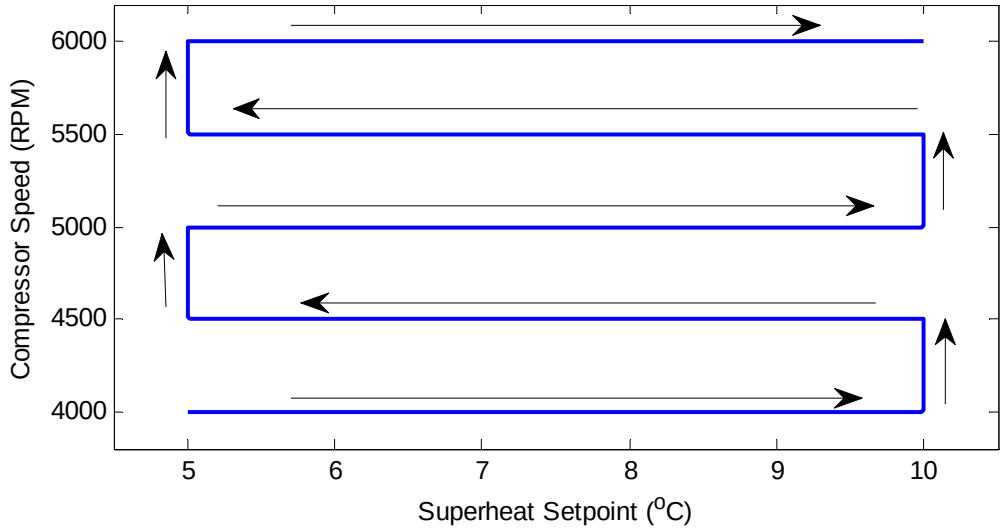


Figure 4.6: Compressor/EEV walkthrough pattern. This is repeated for several different water flow rates through the evaporators to get a complete compressor map.

The following polynomial formula for water pump power was used:

$$\dot{w}_{wp} = 0.275 - 0.15u_{wp} + 0.025u_{wp} \tag{4.12}$$

This will take the place of fan work in the ICOP equation, since the experimental system uses water pumps instead of air fans. The input signal u_{wp} represents the voltage sent to the DAQ system to control the pump speed, and ranges from 2.25 to 5V. This corresponds to a voltage supply at the pump of 6 to 12 V, with sufficient current to drive

the pump (approximately 1.5 amperes at maximum). Note that this formula is assumed for the purposes of demonstration of the architecture and does not correlate with the actual power draw of the pump.

The data, plus the defined fan power, is fed into Equation (4.4). A least squares fit results in the following formula for ICOP at time *k*:

$$\beta^{-1}{}_k = \begin{bmatrix} p_k \\ q_{k,total} \\ sh_k \end{bmatrix}^T \begin{bmatrix} 0.00 & 0.002 & 0.002 \\ 0.002 & 0.0853 & 0.035 \\ 0.002 & 0.035 & 0.0292 \end{bmatrix} \begin{bmatrix} p_k \\ q_{k,total} \\ sh_k \end{bmatrix} + \begin{bmatrix} -0.152 \\ -2.34 \\ -2.19 \end{bmatrix}^T \begin{bmatrix} p_k \\ q_{k,total} \\ sh_k \end{bmatrix} + 40.86 \quad (4.13)$$

Using the gains given in Table 4.1, the cost function solved by the centralized controller becomes:

$$J = \sum_{j=0}^{Nu} \left(\sum_{i=1}^{n} \begin{Bmatrix} {}^{i}u_{k+j}{}^{T} \left({}^{i}H^{T}\, {}^{i}\Lambda_e\, {}^{i}H + {}^{i}\Lambda_\Delta \right) {}^{i}u_{k+j} \\ +2({}^{i}T_k\, {}^{i}F^{T}\, {}^{i}\Lambda_e\, {}^{i}H + {}^{i}d_k\, {}^{i}K^{T}\, {}^{i}\Lambda_e\, {}^{i}H \\ -{}^{i}T_{set}\, {}^{i}\Lambda_e\, {}^{i}H + {}^{i}U_{k-1}{}^{T}\, {}^{i}\Lambda_\Delta) {}^{i}u_{k+j} \end{Bmatrix} + \beta^{-1}{}_{k+j} \right) \quad (4.14)$$

Table 4.1: Supervisory Controller Parameters

Constraints

Signal	Minimum Value	Maximum Value
Superheat (°C)	**6**	**12**
Cooling per Evaporator (kW)	**0.55**	**1.2**
Total Cooling (kW)	**n/a**	**3.5**
Suction Pressure (kPa)	**300**	**425**

Controller Parameters

Sample Time (s)	**300**
Control Horizon (time steps)	**5**
Prediction Horizon (time steps)	**15**

Weights

Error, Λ_e	**0.0075 (baseline)**
Slew rate, Λ_Δ	**0.0005**
ICOP, Λ_φ	**1**

Local MPC Controllers

The local MPC controllers were implemented using the MatLab MPC Toolbox [115]. Each controller is assigned to an evaporator, which is modeled as a two-input, two-output discrete linear time invariant plant. The inputs for each plant are the EEV position (*v*) and evaporator water pump speed (*f*), and the outputs are cooling (*q*) and superheat (*sh*). The dynamic models are experimentally identified using linear system identification techniques [112] and take the following form:

$$\begin{aligned} {}^{i}x_{k+1} &= {}^{i}A\,{}^{i}x_k + {}^{i}B\begin{bmatrix} v_k \\ f_k \end{bmatrix} \\ {}^{i}y_k &= \begin{bmatrix} {}^{i}q_k \\ {}^{i}sh_k \end{bmatrix} = {}^{i}C\,{}^{i}x_k \end{aligned} \tag{4.15}$$

The matrices for the three evaporator models are given below:

$$ {}^{1}A = \begin{bmatrix} 0.86 & -0.11 \\ -0.27 & 0.76 \end{bmatrix}, {}^{1}B = \begin{bmatrix} 0.02 & -0.001 \\ 0.04 & 0.002 \end{bmatrix}, {}^{1}C = \begin{bmatrix} 0.53 & 0.48 \\ -8.60 & 1.61 \end{bmatrix} \tag{4.16} $$

$$ {}^{2}A = \begin{bmatrix} 0.76 & -0.07 \\ -0.08 & 0.96 \end{bmatrix}, {}^{2}B = \begin{bmatrix} 0.01 & 0.002 \\ 0.004 & 0.001 \end{bmatrix}, {}^{2}C = \begin{bmatrix} 0.65 & 0.25 \\ -8.33 & 18.28 \end{bmatrix} \tag{4.17} $$

$$ {}^{3}A = \begin{bmatrix} 0.70 & -0.26 \\ -0.17 & 0.82 \end{bmatrix}, {}^{3}B = \begin{bmatrix} 0.04 & -0.002 \\ 0.03 & 0.0003 \end{bmatrix}, {}^{3}C = \begin{bmatrix} 0.48 & 0.56 \\ -11.18 & 14.35 \end{bmatrix} \tag{4.18} $$

The controller minimizes the following cost function, modified from [115] to reflect the actual inputs and outputs. Note that no weight is placed upon the superheat error. This means that the controller will ignore the superheat setpoint unless the model predicts that superheat will exceed the specified constraints.

$$ \begin{aligned} J(k) = & \sum_{j=1}^{Ny} \lambda_q \left(qset_{k+j} - q_{k+j} \right)^2 \\ & + \sum_{j=1}^{Nu} \left\{ \lambda_v \left(v_{k+j} - v_{k+j-1} \right)^2 + \lambda_f \left(f_{k+j} - f_{k+j-1} \right)^2 \right\} \end{aligned} \tag{4.19} $$

$$ \begin{aligned} sh_{min} \le sh_k \le sh_{max} \\ v_{min} \le v_k \le v_{max} \\ f_{min} \le f_k \le f_{max} \end{aligned} $$

This control law meets the requirements of the architecture; it tracks a cooling setpoint (weighted by λ_Q) generated by the supervisory controller and keeps superheat in a band; additionally, weights are placed to keep the EEV and fan actuators (λ_v and λ_f, respectively) from moving too quickly, unless the superheat constraint is violated. The controller parameters, weights, and constraints for the local controllers are given in Table 4.2.

Table 4.2: Local Controller Details

Constraints

Signal	Minimum Value	Maximum Value
Superheat (°C)	**6**	**12**
Cooling (kW)	**0.5**	**1.2**
EEV (%)	**0**	**100**
WP (%)	**0**	**100**

Controller Parameters

Sample Time (s)	**8**
Control Horizon (time steps)	**3**
Prediction Horizon (time steps)	**12**

Weights

Cooling, λ_q	**1**
Superheat, λ_{sh}	**0**
Change in EEV, λ_v	**0.2**
Change in WP, λ_f	**0.2**

Experimental Results

Baseline Test

For the first test shown, the entire control system was run to bring the water temperature in the three zones down to a desired setpoint as shown. During this test, an immersion heater was used to inject a disturbance into the plant, which simulates the load that a room or conditioned space would encounter due to lights, electronics, people, etc.

Figure 4.7 displays some of the functionality of the control architecture. At the beginning of the experimental run, the zone errors are all approximately 3 °C; as the water is chilled, the error decreases. As this occurs, the cooling setpoints are decreased by the supervisor, resulting in an asymptotic approach to a final steady state temperature. Note that all zones display a steady state error, which the supervisor controller accepts as a tradeoff between zone setpoint tracking and energy consumption. The cooling also approaches the value of the disturbance by the end of the test, allowing steady state temperatures to be reached. Figure 4.8 shows the superheat during this experimental run. Note that the local evaporator controllers do not attempt to track the setpoint, but allow the superheat to float in a band of 6 to 12 °C as required.

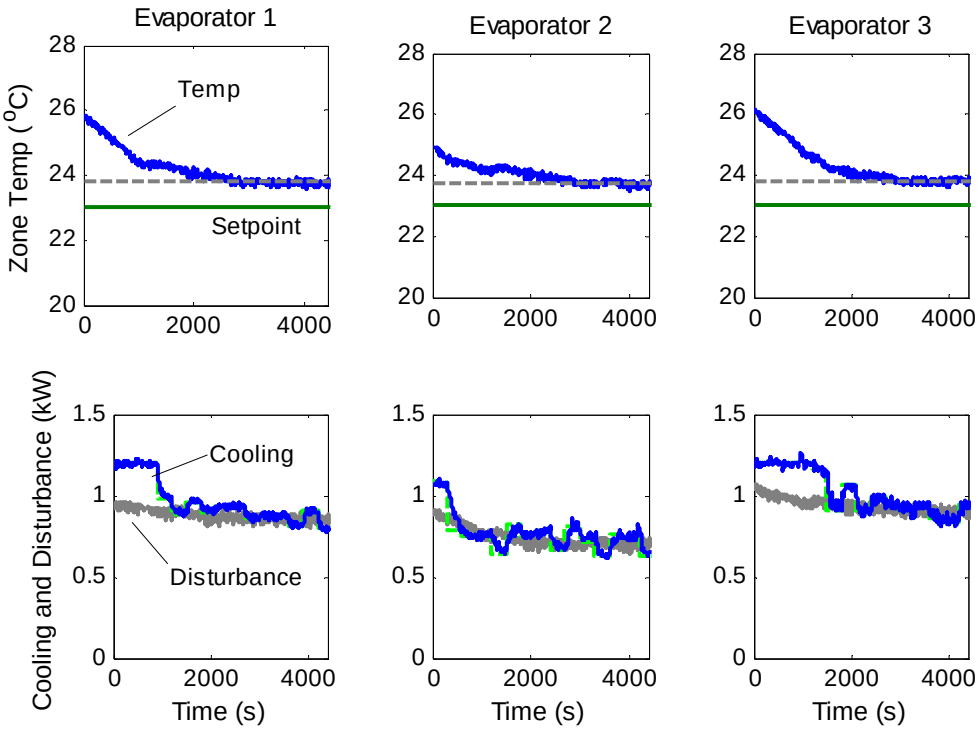


Figure 4.7: Baseline test temperatures, cooling, and disturbances

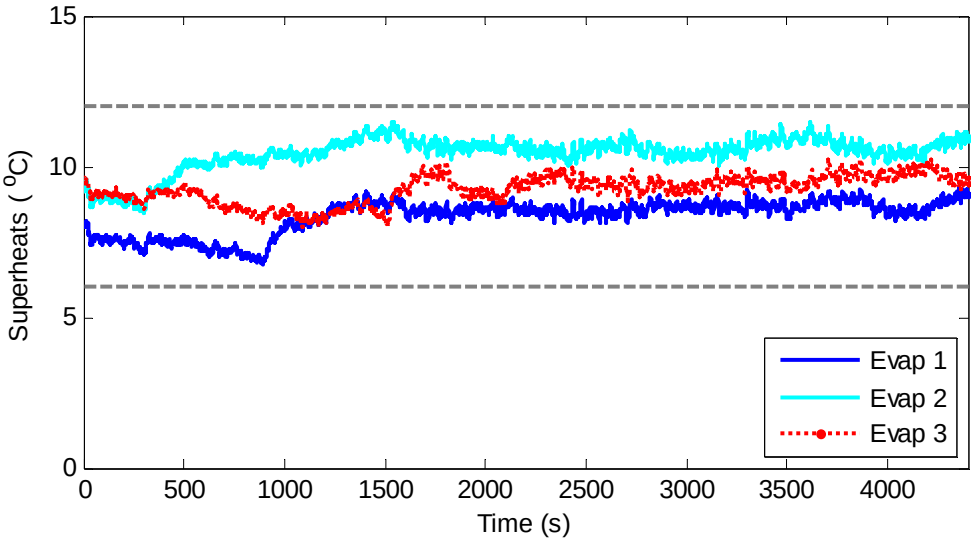


Figure 4.8: Baseline test superheats

Setpoint Tracking and Disturbance Rejection

The following figures demonstrate the setpoint tracking and disturbance rejection characteristics of the supervisory controller. Figure 4.9 shows how the system can successfully increase the zone temperature as well as decrease it as shown earlier; in the event of a step increase in setpoint, the controller brings the zone's cooling to the minimum allowable; since this is smaller in magnitude than the disturbance, the zone's temperature increases to a new steady state condition; furthermore, the steady state temperature is still higher than the setpoint rather than the lower; this is a result of the controller seeking to maximize energy efficiency. Otherwise, the supervisor might allow the system to settle at the same error, but lower than the setpoint.

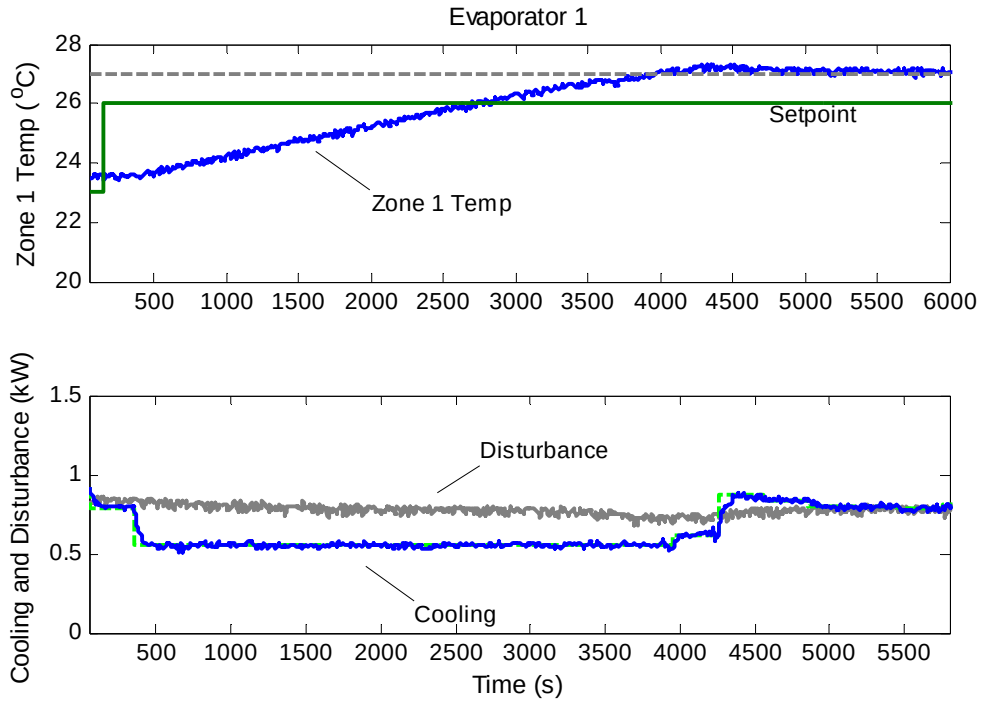


Figure 4.9: Setpoint increase

For the test in Figure 4.10, an additional disturbance was added to the zone (i.e., another heater was turned on). As the disturbance estimator notes this and informs the supervisor, the supervisor re-calculates the new optimum, resulting in a larger tolerated steady state error for the same setpoint and system conditions.

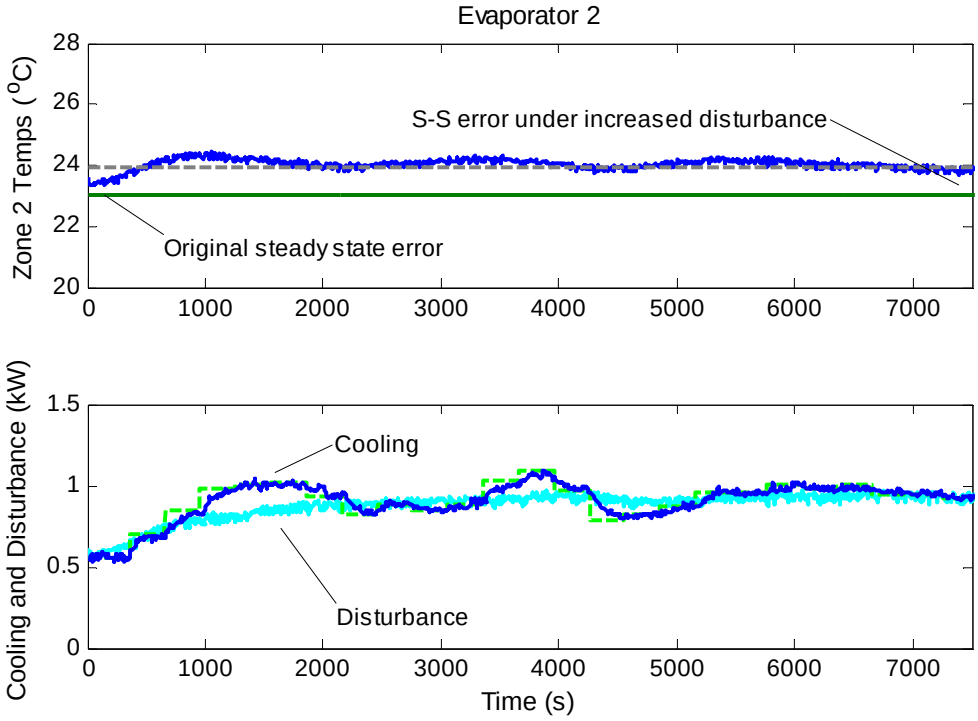


Figure 4.10: Disturbance increase

Figure 4.11 also shows how a step decrease in the setpoint results in the supervisor permitting more cooling to the evaporator in question, until the same steady state error is reached above the new setpoint. The cooling and superheat are presented as well; note how the local MPC controller performs a demand shedding function by not always meeting the cooling setpoint after 1500s, since to do so would require the EEV to open further, decreasing superheat past the allowable value.

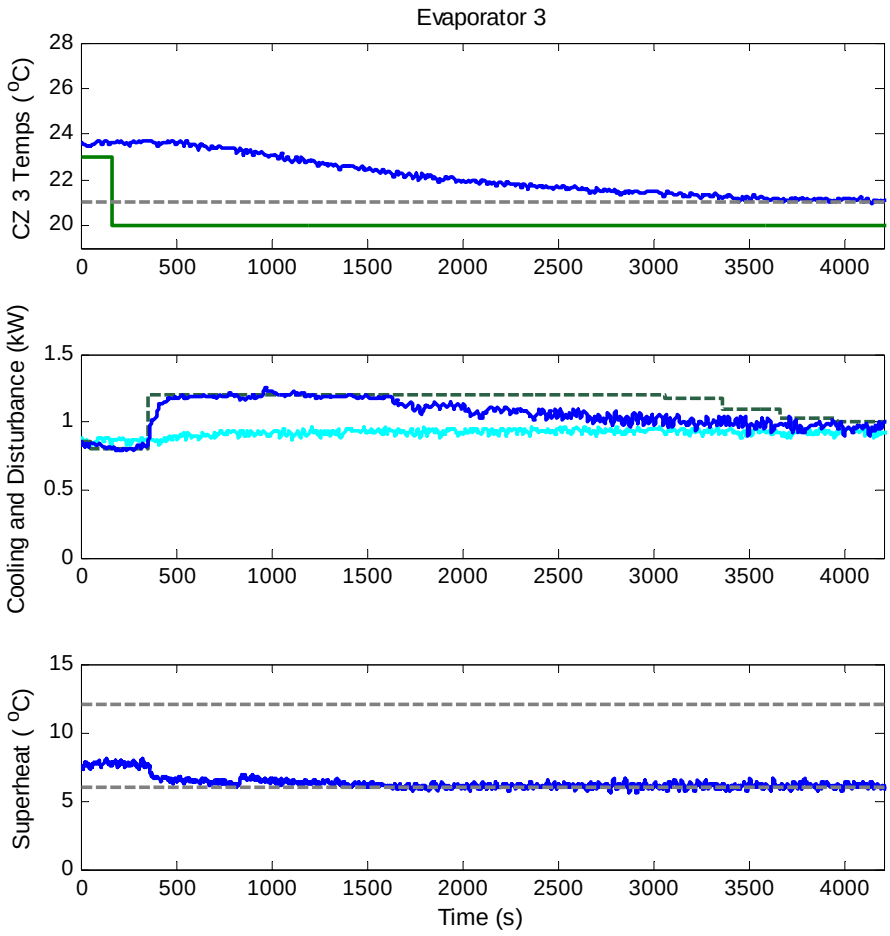


Figure 4.11: Setpoint step decrease

Efficiency Weights in Supervisory Controller

As noted earlier, the supervisory controller in Test 1 allowed a steady state error of approximately 0.75 °C in the cooling zone temperature. Two more tests were run using the same test conditions but with different weights upon the error. Figure 4.12 shows the zone 3 temperature for each of the three runs superimposed on each other for comparison. As the weight on tracking error increases, the supervisor permits a smaller steady state error.

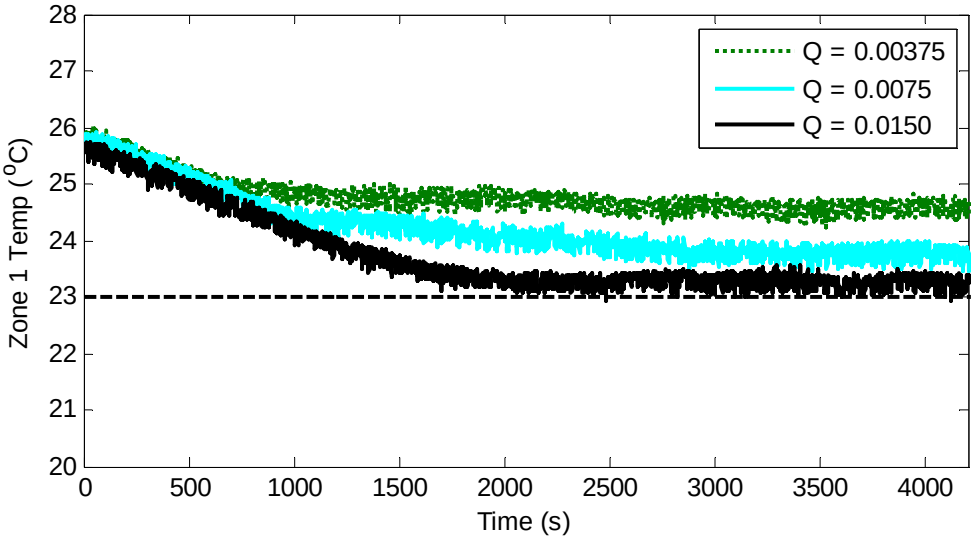


Figure 4.12: Steady state errors for three different error weights

Figure 4.13 shows the instantaneous power consumption of the system and the cumulative energy consumption during the three tests; the higher the error weight, the compressor tends to spin faster (to provide more cooling) and consumes more power.

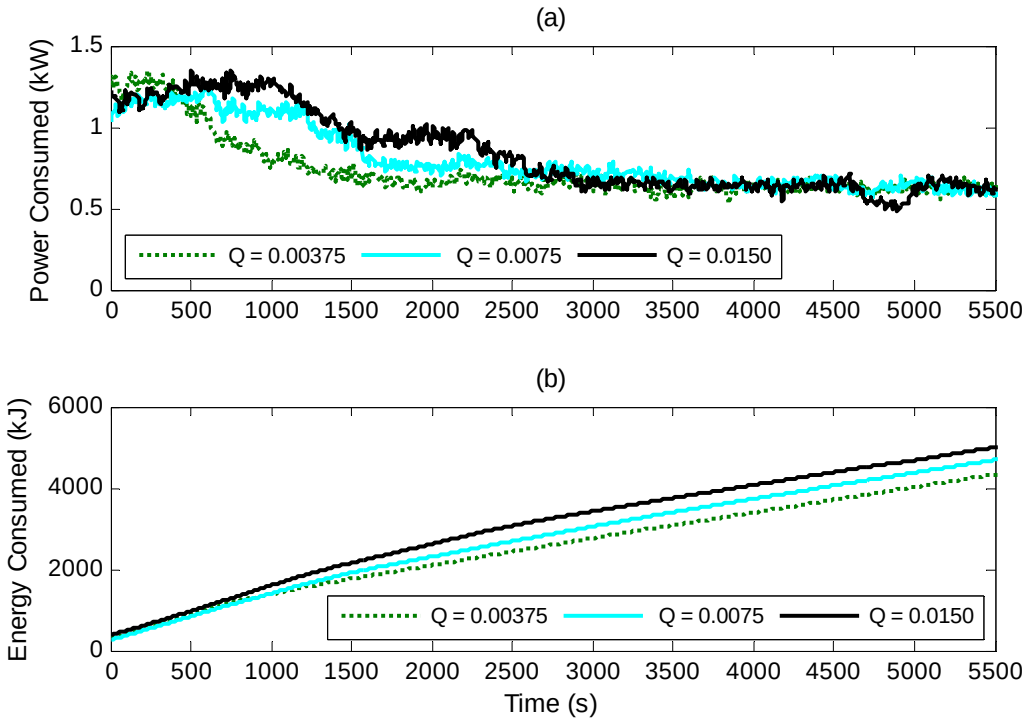


Figure 4.13: Power Consumption and Energy Consumption

Energy Efficiency

Earlier tests have shown that the supervisor successfully brings zone temperatures to a temperature near the setpoint, balancing the desires of the user with maximizing system operating efficiency. To examine the behavior of the supervisor from another perspective, a test was run to allow the controller to move the system from a randomly chosen operating point to a more energy efficient one, holding the temperature error constant. At the start of this test, the zones were allowed to come to the steady state temperature errors associated with the error weights, although the compressor speed is tracking a random setpoint. Then the supervisory controller was activated at approximately 190 seconds, which calculated an optimal pressure setpoint for the compressor to track. Since the zones were already at the steady state temperature, the cooling rates and disturbance are already matched and the controller didn't change them. As the supervisor seeks to optimize the setpoints, it brings the system to a more efficient operating condition, as shown in Figure 4.14.

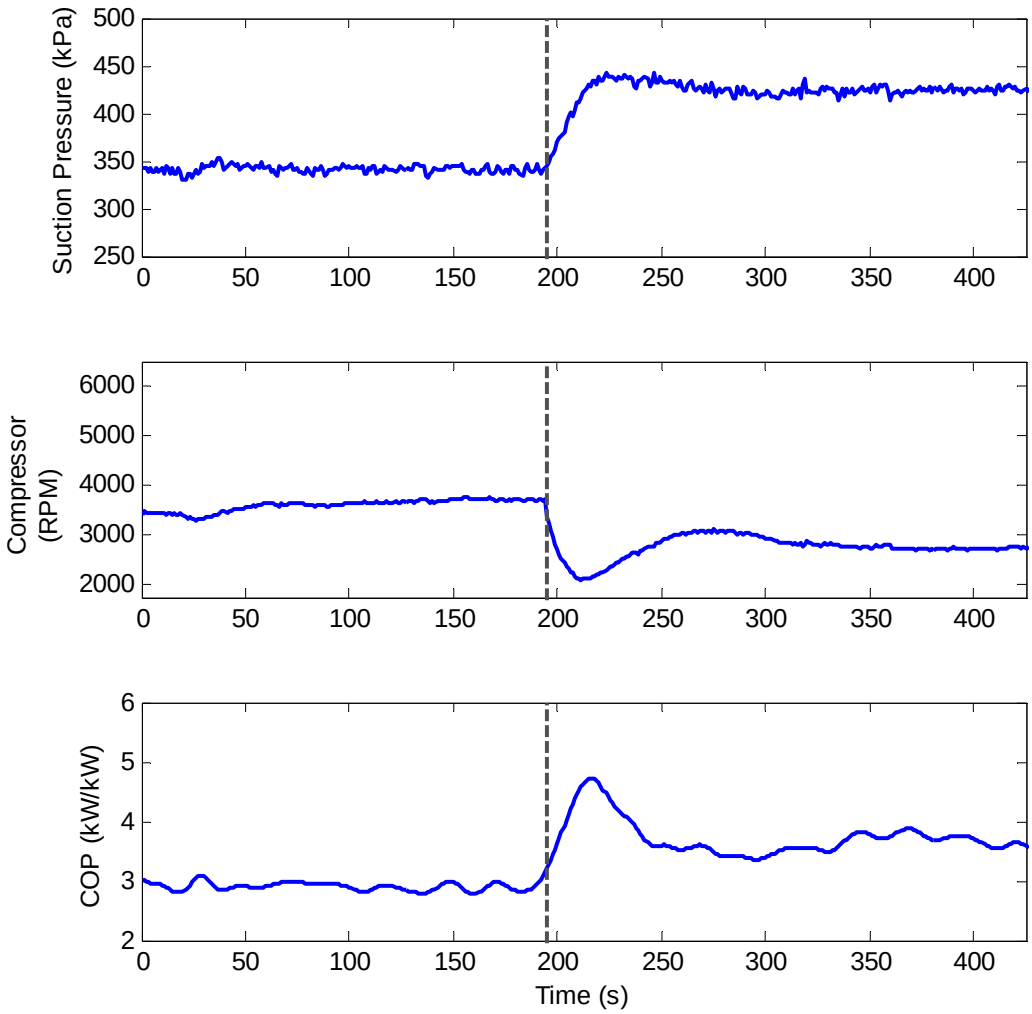


Figure 4.14: Efficiency seeking behavior

This chapter refined a novel control architecture for multiple evaporator cooling systems. The architecture is a flexible, two-level structure that uses model predictive control methods at both levels to achieve the respective goals of each controller. The supervisory controller seeks to regulate the temperatures of different zones serviced by the same vapor compression cycle, while maximizing energy efficiency in the process.

This control architecture, however, still requires a supervisory controller with knowledge of the system as a whole. The next step in successfully distributing the control of HVAC systems is for a control architecture that successfully achieves a system-level optimum through communication between individual agents, rather than a supervisor. A control approach is presented in the next chapter that meets this description.

CHAPTER V

THEORY OF NEIGHBOR COMMUNICATION OPTIMIZATION

The research presented in this dissertation is motivated in part by a need for effective control of the large, spatially distributed networks of components that comprise building HVAC&R systems. These systems generally feature a suite of local level dynamic controllers that control individual components to track setpoints generated by agents acting further up a hierarchy. This is very similar to the structure frequently found in chemical plants, where higher-level optimizers generate setpoints for valves and other equipment to achieve commercially optimal operations. The work presented in this chapter is a novel method of calculating setpoints that achieve an overall, centralized optimum without a centralized optimizer. Instead, the optimization is distributed among local subsystem optimizers. These optimizers have knowledge of their own subsystem only, and communicate costs and outputs to their immediate neighbors only. By judicious construction of the local cost functions, the optimizers can iterate to the steady state conditions that correspond to the minimal cost of a system-wide cost function. The method builds upon distributed MPC and leverages the structure frequently found in HVAC systems, wherein the output of one plant acts as a disturbance to others.

The rest of the chapter is organized as follows. Mathematical preliminaries and definitions of the various terms used throughout the chapter are presented. Then the cost functions for the local controllers are given, and the algorithm is detailed. Proofs for

stability of the optimization and convergence to the centralized optimum are presented. Examples of the application of the algorithm will be presented in following chapters.

Preliminaries

For the purposes of this discussion, two plants or subsystems are "neighbors" if the output of one acts as a disturbance to the other. The "upstream" neighbor disturbs the "downstream" neighbor. See Figure 5.1. Each of the plants is assumed to be described as a linear time invariant, discrete time dynamical system:

$$\begin{aligned} x_i^+ &= A_i x_i + B_{u,i} u_i + B_{v,i} v_i + B_{d,i} d_i \\ y_i &= C_{y,i} x_i \\ z_i &= C_{z,i} x_i \end{aligned} \tag{5.1}$$

With state variables x_i, control inputs u_i, inputs from upstream systems v_i, regulated outputs y_i, and outputs to downstream systems z_i. Additionally, reference signals are denoted r_i and error signals are defined as:

$$e_i = (r_i - y_i) \tag{5.2}$$

For notational simplicity, the superscript + will be used to signify the value at the next iteration or sample, e.g., time $k+1$, where k is the current sampling instant.

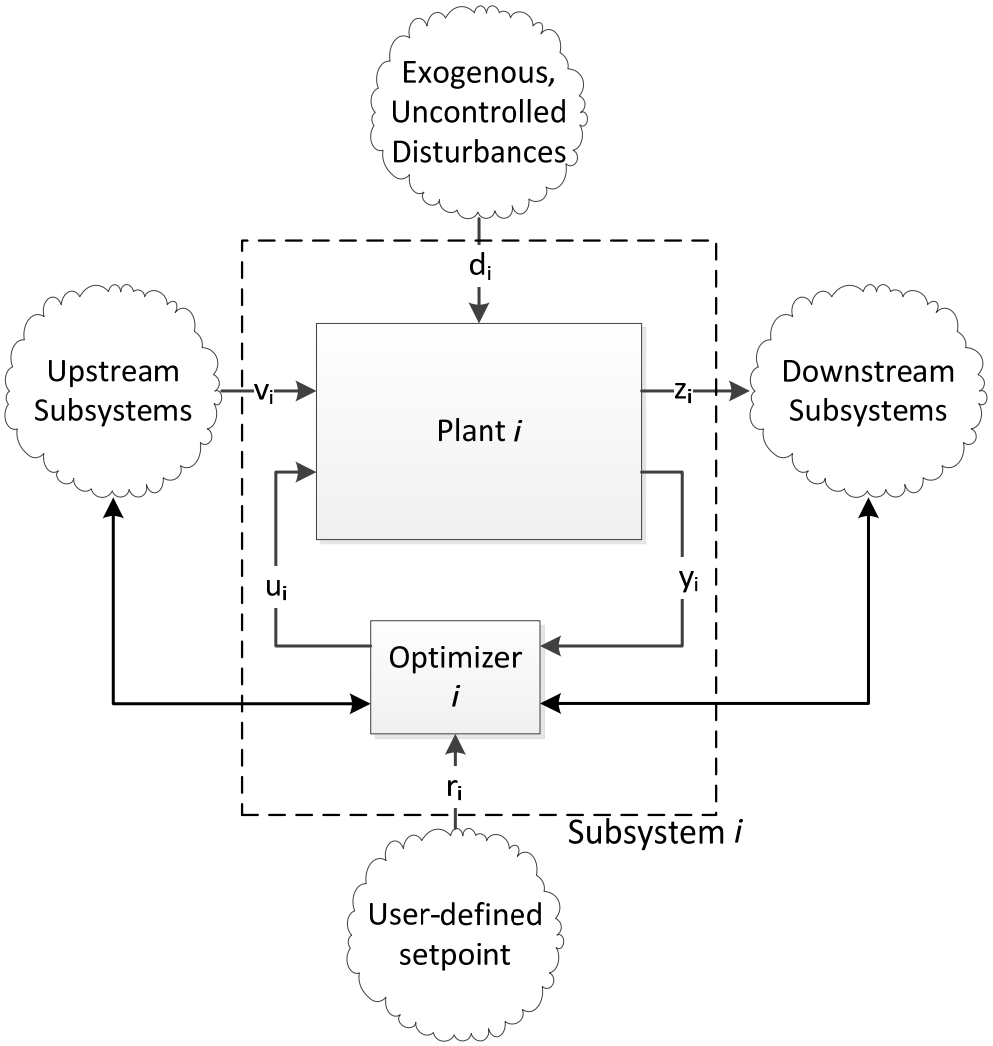


Figure 5.1: Upstream and downstream plants.

The Neighbor Communication Optimization (NC-OPT) architecture calculates the predicted steady state optimum for each of the plants; in the following discussion an over-bar will be used to indicate the steady state value of a discrete-time signal, e.g. $\overline{y_i}$. The predicted steady state outputs for the i^{th} plant are thus:

$$\begin{aligned}\overline{y}_i &= \left[C_{y,i}\left(I-A_i\right)^{-1}B_{u,i}\right]\overline{u}_i + \left[C_{y,i}\left(I-A_i\right)^{-1}B_{v,i}\right]\overline{v}_i + \left[C_{y,i}\left(I-A_i\right)^{-1}B_{d,i}\right]\overline{d}_i \\ &= M_{y,i}\overline{u}_i + N_{y,i}\overline{v}_i + P_{y,i}\overline{d}_i\end{aligned} \tag{5.3}$$

$$\begin{aligned}\overline{z}_i &= \left[C_{z,i}\left(I-A_i\right)^{-1}B_{u,i}\right]\overline{u}_i + \left[C_{z,i}\left(I-A_i\right)^{-1}B_{v,i}\right]\overline{v}_i + \left[C_{z,i}\left(I-A_i\right)^{-1}B_{d,i}\right]\overline{d}_i \\ &= M_{z,i}\overline{u}_i + N_{z,i}\overline{v}_i + P_{z,i}\overline{d}_i\end{aligned} \tag{5.4}$$

For a network consisting of p interconnected systems, define the following stacked vectors.

Control Inputs:

$$\overline{u} = \left[\overline{u}_1^T \quad \ldots \quad \overline{u}_p^T\right]^T \tag{5.5}$$

Inputs from Upstream Systems:

$$\overline{v} = \left[\overline{v}_1^T \quad \ldots \quad \overline{v}_p^T\right]^T \tag{5.6}$$

Regulated Outputs:

$$\overline{y} = \left[\overline{y}_1^T \quad \ldots \quad \overline{y}_p^T\right]^T \tag{5.7}$$

Exogenous, Uncontrolled Disturbances:

$$\overline{d} = \left[\overline{d}_1^T \quad \ldots \quad \overline{d}_p^T\right]^T \tag{5.8}$$

Reference Signals:

$$r = \left[r_1^T \quad \ldots \quad r_p^T\right]^T \tag{5.9}$$

Error Signals:

$$\bar{e} = \begin{bmatrix} \bar{e}_1^T & \dots & \bar{e}_p^T \end{bmatrix}^T \tag{5.10}$$

Outputs to Downstream Systems:

$$\bar{z} = \begin{bmatrix} \bar{z}_1^T & \dots & \bar{z}_p^T \end{bmatrix}^T \tag{5.11}$$

Define the following block diagonal matrices:

$$\mathbf{M_y} = \begin{bmatrix} M_{y,1} & & 0 \\ & \ddots & \\ 0 & & M_{y,p} \end{bmatrix}, \quad \mathbf{N_y} = \begin{bmatrix} N_{y,1} & & 0 \\ & \ddots & \\ 0 & & N_{y,p} \end{bmatrix}, \quad \mathbf{P_y} = \begin{bmatrix} P_{y,1} & & 0 \\ & \ddots & \\ 0 & & P_{y,p} \end{bmatrix} \tag{5.12}$$

$$\mathbf{M_z} = \begin{bmatrix} M_{z,1} & & 0 \\ & \ddots & \\ 0 & & M_{z,p} \end{bmatrix}, \quad \mathbf{N_z} = \begin{bmatrix} N_{z,1} & & 0 \\ & \ddots & \\ 0 & & N_{z,p} \end{bmatrix}, \quad \mathbf{P_z} = \begin{bmatrix} P_{z,1} & & 0 \\ & \ddots & \\ 0 & & P_{z,p} \end{bmatrix} \tag{5.13}$$

Thus we can define the entire system's steady state condition with the following relationships:

$$\begin{aligned} \bar{y} &= \mathbf{M_y}\bar{u} + \mathbf{N_y}\bar{v} + \mathbf{P_y}\bar{d} \\ \bar{z} &= \mathbf{M_z}\bar{u} + \mathbf{N_z}\bar{v} + \mathbf{P_z}\bar{d} \end{aligned} \tag{5.14}$$

In order to capture the interconnections between the plants, define an interconnection matrix Φ of size $n_v \times n_z$, where n_v and n_z are the sizes of the $\bar{v}$ and $\bar{z}$ vectors, respectively:

$$\bar{v} = \Phi\bar{z} \tag{5.15}$$

The interconnections are defined such that $n_v = n_z$. Since every output $\bar{z}_i$ is a disturbance $\bar{v}_i$ to one other plant, the interconnection matrix consists of columns with a

single non-zero entry equal to 1 in each column. Thus the columns are an orthonormal basis in $\mathbb{R}^{n_v}$, making Φ an orthogonal matrix, so $\Phi^T = \Phi^{-1}$. This matrix defines the following steady state relationships:

$$\begin{aligned}\overline{v} &= \Phi\overline{z} \\ &= \left(I - \Phi\mathbf{N_z}\right)^{-1}\Phi\mathbf{M_z}\overline{u} + \left(I - \Phi\mathbf{N_z}\right)^{-1}\Phi\mathbf{P_z}\overline{d} \\ &= \mathbf{W}\overline{u} + \mathbf{X}\overline{d}\end{aligned} \tag{5.16}$$

The local optima are determined by quadratic cost functions. These functions place weight upon the error ($\overline{e_i}$) and input selected by the optimizer ($\overline{u_i}$) using positive symmetric matrices Q_i and S_i, respectively:

$$J_i = \overline{e}_i^T Q_i \overline{e}_i + \overline{u}_i^T S_i \overline{u}_i \tag{5.18}$$

Stacking the local cost function matrices gives the centralized weight matrices:

$$\mathbf{Q} = \begin{bmatrix} Q_1 & & 0 \\ & \ddots & \\ 0 & & Q_n \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} S_1 & & 0 \\ & \ddots & \\ 0 & & S_n \end{bmatrix} \tag{5.19}$$

This allows definition of the centralized cost function:

$$J = \overline{e}^T \mathbf{Q} \overline{e} + \overline{u}^T \mathbf{S} \overline{u} \tag{5.20}$$

This is the cost function that defines the system-level optimum. The operating conditions that yield a minimum of this function is referred to as the "centralized optimum" in the following chapters. Note that even though the cost matrices are block diagonal, optimizing each individual cost independently will not yield the minimum, since interconnections must be taken into account in order to find this optimum. This idea is explored in more detail in the next section.

Neighbor Communication Optimization (NC-OPT)

Centralized and Decentralized Cost Functions

Equation (5.20) defines the centralized cost function whose minimum is the centralized or "Pareto" optimum. The objective of the NC-OPT architecture is to achieve this optimum with a distributed set of optimizers that have no knowledge of the other plants' dynamics, and only communicate with their immediate upstream and downstream neighbors. Substituting the steady state prediction equations (5.14) and (5.16) into the centralized cost function (5.20) yields:

$$\begin{aligned} J &= \bar{e}^T \mathbf{Q} \bar{e} + \bar{u}^T \mathbf{S} \bar{u} \\ &= \bar{u}^T \left[\mathbf{S} + \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right)^T \mathbf{Q} \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right) \right] \bar{u} \\ &\quad + 2 \left[\left(-r^T + \bar{d}^T \left(\mathbf{P}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{X} \right)^T \right) \mathbf{Q} \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right) \right] \bar{u} \\ &\quad + \left[\left(\left(\mathbf{P}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{X} \right) \bar{d} - r \right)^T \mathbf{Q} \left(\left(\mathbf{P}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{X} \right) \bar{d} - r \right) \right] \end{aligned} \tag{5.21}$$

In general, distributed optimizers will reach an equilibrium position that is not on the Pareto optimal surface. In order for distributed optimizers to achieve the centralized optimum, each optimizer must solve the same cost function, and must have accurate information about the actions of the other optimizers to account for cross-coupling effects. As noted in the literature review, earlier distributed model predictive control (MPC) approaches use controllers with an interaction model of all the other plants and communication with every other controller, iterating to the Pareto solution.

The central idea in the NC-OPT approach is that every plant only communicates with its immediate neighbor, and each optimizer has no knowledge of any plant other than its own. Each optimizer communicates to its downstream neighbors its locally

optimal output; additionally, it communicates to its upstream neighbors the additional operating cost created by the upstream disturbances (which are outputs of the upstream plants). Since, after sufficient iteration, the local plants will take into account the effects of their own actions on the network as a whole, the optimizers are solving the centralized cost function and the Pareto optimum will be reached. In practice, the optimizers will often be paired with local component dynamic controllers; thus, the NC-OPT inputs calculated by these optimizers will be setpoints for the lower level controllers to meet. The local cost function that is minimized for the i^{th} plant is as follows:

$$\begin{aligned} J_i &= \bar{e}_i^T Q_i \bar{e}_i + \bar{u}_i^T S_i \bar{u}_i + \psi_i^T \bar{z}_i \\ &= \left[r_i^T Q_i r_i + \bar{d}_i^T P_{y,i}^T Q_i P_{y,i} \bar{d}_i + \psi_i^T P_{z,i} \bar{d}_i - 2 r_i^T Q_i P_{y,i} \bar{d}_i \right] + \bar{u}_i^T \left[M_{y,i}^T Q_i M_{y,i} + S_i \right] \bar{u}_i \\ &\quad + \left[-2 r_i^T Q_i M_{y,i} + 2 \bar{d}_i^T P_{y,i}^T Q_i M_{y,i} + \psi_i^T M_{z,i} \right] \bar{u}_i \\ &\quad + \begin{Bmatrix} \bar{v}_i^T \left(N_{y,i}^T Q_i N_{y,i} \right) \bar{v}_i \\ + \left(2 \bar{d}_i^T P_{y,i}^T Q_i N_{y,i} + 2 \bar{u}_i^T M_{y,i}^T Q_i N_{y,i} - 2 r_i^T Q_i N_{y,i} + \psi_i^T N_{z,i} \right) \bar{v}_i \end{Bmatrix} \end{aligned} \tag{5.22}$$

The first two terms of (5.22) comprise the standard decentralized cost function, with knowledge of disturbances coming from upstream (refer to (5.1)). The term ψ_i represents the costs imposed on the immediate downstream plants by the i^{th} plant's outputs $\bar{z}_i$. These costs are communicated to it by the downstream neighbors. Equation (5.22) is the cost function that the local, distributed optimizers minimize. The increase in cost of this function due to a change of the upstream disturbances is designated γ_i, a linear function of $\bar{v}_i$, $\bar{u}_i$, and ψ_i:

$$\gamma_i = \frac{\partial J_i}{\partial \bar{v}_i} = 2N_{y,i}{}^T Q_i M_{y,i} \bar{u}_i + 2N_{y,i}{}^T Q_i N_{y,i} \bar{v}_i + N_{z,i} \psi_i - 2N_{y,i}{}^T Q_i r_i + 2N_{y,i}{}^T Q_i P_{y,i} \bar{d}_i \quad (5.23)$$

This is the term that each local optimizer will pass to its upstream neighbors to inform them of the costs imposed by their actions. The distribution of downstream costs is described using the same Φ matrix defined earlier, thus:

$$\psi = \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_p \end{bmatrix} = \Phi^T \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_p \end{bmatrix} \quad (5.24)$$

The algorithm is detailed next.

Algorithm

The global optimization is distributed among the individual plants; in order to achieve a global optimum, an iterative approach is required. At predetermined time intervals (optimizer sampling rate), the algorithm proceeds thus for a given number of iterations, c_{max}:

Step 0. Initialize variables for optimizer iteration (not plant) step *j=0*. Each optimizer assigns initial values to $\bar{v}_i$, $\bar{u}_i$, and ψ_i; these are generally the last held value from the previous set of iterations. If this is the very first calculation (i.e., the plant's *t=0* as well), these variables are set to zero.

Step 1. (Iterations start here). Each optimizer communicates its stored $\overline{z}_i$ value to its downstream neighbor; this becomes $\overline{v}_i$ for the downstream plant. Similarly, pass the value stored in each γ_i to the upstream plants, which becomes ψ_i for the upstream neighbors. This creates the following update laws:

$$\psi^+ = \Phi^T \gamma \tag{5.25}$$

$$\overline{v}^+ = \Phi \overline{z} \tag{5.26}$$

Step 2. Each optimizer calculates the optimal solution to its cost function $u_i^{QP} = \arg\min(J_i)$ per (5.22).

Step 3. Calculate each updated optimum $\overline{u}_i^+$ as a convex combination of u_i^{QP} and the previous value of $\overline{u}_i$. This is the $\overline{u}_i$ for iteration step *j+1*. For tuning parameter $w \in [0,1)$:

$$\overline{u}_i^+ = w\overline{u}_i + (1-w)u_i^{QP} \tag{5.27}$$

Step 4. Calculate the predicted outputs $\overline{z}^+$ using the new $\overline{u}_i^+$ (iteration *j+1*) and the last known $\overline{v}_i$ (from iteration *j*):

$$\overline{z}_i^+ = M_{z,i}\overline{u}_i^+ + N_{z,i}\overline{v}_i + P_{z,i}\overline{d}_i \tag{5.28}$$

Step 5. Calculate the γ terms to pass to upstream neighbors. From equation (5.23):

$$\gamma_i^+ = 2N_{y,i}{}^T Q_i M_{y,i} \bar{u}_i^+ + 2N_{y,i}{}^T Q_i N_{y,i} \bar{v}_i + N_{z,i}{}^T \psi_i - 2N_{y,i}{}^T Q_i r_i + 2N_{y,i}{}^T Q_i P_{y,i} \bar{d}_i \qquad (5.29)$$

Step 6. Check iteration count. If the maximum number of iterations has been reached, stop and progress to step 7. If not, repeat steps 1 through 6.

Step 7. Apply the final values of the calculated $\bar{u}_i$ terms to the plants. The values of $\bar{u}_i$ might be actuator positions, but most likely will be setpoints passed to lower-level dynamic controllers. Repeat process at predetermined intervals (optimizer sampling rate).

Stability and Convergence of Information Dynamics

Stability

The dynamics of the information passed between controllers as described above can be modeled as a discrete-time state space system with the setpoint vector *r* as a disturbance, along with exogenous disturbances $\bar{d}$ and the (bounded) results of the optimizers $u^{QP,NC}$ as the "control" input. From equations (5.25)-(5.29):

$$\begin{bmatrix} \bar{u}^+ \\ \bar{v}^+ \\ \psi^+ \end{bmatrix} = \begin{bmatrix} wI & 0 & 0 \\ w\Phi\mathbf{M_z} & \Phi\mathbf{N_z} & 0 \\ 2w\Phi^T\mathbf{N_y}^T\mathbf{QM_y} & 2\Phi^T\mathbf{N_y}^T\mathbf{QN_y} & \Phi^T\mathbf{N_z}^T \end{bmatrix} \begin{bmatrix} \bar{u} \\ \bar{v} \\ \psi \end{bmatrix} + \begin{bmatrix} (1-w)I \\ (1-w)\Phi\mathbf{M_z} \\ 2(1-w)\Phi^T\mathbf{N_y}^T\mathbf{QM_y} \end{bmatrix} \begin{bmatrix} u^{QP,NC} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -2\Phi^T\mathbf{N_y}^T\mathbf{Q} \end{bmatrix} \begin{bmatrix} r \end{bmatrix} + \begin{bmatrix} 0 \\ \Phi\mathbf{P_z} \\ 2\Phi^T\mathbf{N_y}^T\mathbf{QP_y} \end{bmatrix} \begin{bmatrix} \bar{d} \end{bmatrix} \tag{5.30}$$

Since the state matrix of the above system is lower block triangular, its eigenvalues are the same as the eigenvalues of the matrices along its diagonal. In order for the above system to be bounded-input bounded-output stable, therefore, the eigenvalues of the matrix $\Phi\mathbf{N_z}$ must lie within the unit circle of the complex plane. This can be physically interpreted as a low-gain requirement from upstream disturbances to outputs going downstream; i.e., $\|\bar{z}/\bar{v}\|$ is "small".

Inside the constrained space the solution to the quadratic programming (QP) problem $\bar{u}_i^{QP}$ can be found by solving $dJ_i/d\bar{u}_i = 0$ for $\bar{u}_i$:

$$\begin{aligned} \bar{u}_i^{QP} &= Y_i M_{y,i}{}^T Q_i \left(r_i - N_{y,i}\bar{v}_i + P_{y,i}\bar{d}_i \right) - 0.5 Y_i M_{z,i}{}^T \psi_i \\ Y_i &= \left(M_{y,i}{}^T Q_i M_{y,i} + S_i \right)^{-1} \end{aligned} \tag{5.31}$$

Since all matrices in (5.31) are block diagonal, the matrices and vectors in the solution (5.31) can be stacked to obtain the following matrix equation:

$$u^{QP,NC} = \mathbf{Y}\mathbf{M}_\mathbf{y}{}^T\mathbf{Q}r - 0.5\mathbf{Y}\mathbf{M}_\mathbf{z}{}^T\psi - \mathbf{Y}\mathbf{M}_\mathbf{y}{}^T\mathbf{Q}\mathbf{N}_\mathbf{y}\bar{v} + \mathbf{Y}\mathbf{P}_\mathbf{y}{}^T\mathbf{Q}\bar{d} \tag{5.32}$$

Substitution of (5.32) into (5.30) and rearranging terms changes the state matrix of the information transfer dynamics within the constraint space to:

$$\begin{bmatrix} wI & -(1-w)\mathbf{Y}\mathbf{M}_\mathbf{y}{}^T\mathbf{Q}\mathbf{N}_\mathbf{y} & -0.5(1-w)\mathbf{Y}\mathbf{M}_\mathbf{z}{}^T \\ w\Phi\mathbf{M}_\mathbf{z} & \Phi\mathbf{N}_\mathbf{z} & 0 \\ 2w\Phi^T\mathbf{N}_\mathbf{y}{}^T\mathbf{Q}\mathbf{M}_\mathbf{y} & 2\Phi^T\mathbf{N}_\mathbf{y}{}^T\mathbf{Q}\mathbf{N}_\mathbf{y} & \Phi^T\mathbf{N}_\mathbf{z}{}^T \end{bmatrix} \tag{5.33}$$

Once again, the eigenvalues of the state matrix dictate the stability of the above system's equilibrium point. Recall that the scalar term *w* is used as a tuning parameter for the algorithm; namely, as *w* approaches 1, the convergence is slowed down. Examination of the state matrix in (5.33) shows that as *w* gets closer to 1, the matrix becomes closer to the state matrix in (5.30). Thus as *w* increases, the eigenvalues can be driven inside the unit circle, ensuring that convergence is possible as long as $\left|\lambda\left(\Phi\mathbf{N}_\mathbf{z}\right)\right| < 1$.

Convergence to Centralized Optimum

The goal of the NC-OPT approach is for the distributed optimizers to converge to the Pareto optimal given a sufficient number of iterations. The preceding section gave conditions on the state matrix of the information dynamic systems for convergence to a stable equilibrium point inside the constrained space of $\bar{u}$; in this section, we show that this equilibrium point is the same as that generated by the centralized optimizer. Recall from (5.21) that the centralized cost function expands to:

$$\begin{aligned} J_c &= \bar{e}^T \mathbf{Q} \bar{e} + \bar{u}^T \mathbf{S} \bar{u} \\ &= \bar{u}^T \left[\mathbf{S} + \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right)^T \mathbf{Q} \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right) \right] \bar{u} \\ &\quad + 2 \left[\left(-r^T + \bar{d}^T \left(\mathbf{P}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{X} \right)^T \right) \mathbf{Q} \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right) \right] \bar{u} \\ &\quad + \left[\left(\left(\mathbf{P}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{X} \right) \bar{d} - r \right)^T \mathbf{Q} \left(\left(\mathbf{P}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{X} \right) \bar{d} - r \right) \right] \end{aligned} \tag{5.34}$$

Calculating $dJ_C / d\bar{u} = 0$ gives a relationship between the optimal $\bar{u}$ (designated $\bar{u}^{OPT}$) and the setpoint *r* and disturbance $\bar{d}$ when the optimum is inside the constraint space:

$$\begin{aligned} &\left(\mathbf{S} + \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right)^T \mathbf{Q} \left(\mathbf{M}_\mathbf{y} + \mathbf{N}_\mathbf{y} \mathbf{W} \right) \right) \bar{u}^{OPT} \\ &= \left(\mathbf{M}_\mathbf{y}{}^T + \mathbf{W}^T \mathbf{N}_\mathbf{y}{}^T \right) \mathbf{Q} \left(r - \left(\mathbf{P}_\mathbf{y}{}^T + \mathbf{X}^T \mathbf{N}_\mathbf{y}{}^T \right) \bar{d} \right) \end{aligned} \tag{5.35}$$

Now examine the cost functions minimized by the NC-OPT optimizers. Recall that calculating $dJ_i / d\bar{u}_i = 0$ gives the following, where $\bar{u}_i^{QP} = \arg\min(J_i)$:

$$\begin{aligned}&\left(M_{y,i}{}^{T}Q_iM_{y,i}+S_i\right)\bar{u}_i^{QP}\\&=\left(M_{y,i}{}^{T}Q_ir_i-0.5M_{z,i}{}^{T}\psi_i-M_{y,i}{}^{T}Q_iN_{y,i}\bar{v}_i-M_{y,i}{}^{T}Q_iP_{y,i}\bar{d}_i\right)\end{aligned}\tag{5.36}$$

Stacking equation (5.36) for all subsystems results in:

$$\begin{aligned}&\left(\mathbf{S}+\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}\mathbf{M}_{\mathbf{y}}\right)\bar{u}^{QP,NC}\\&=\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}r-\frac{1}{2}\mathbf{M}_{\mathbf{z}}{}^{T}\psi-\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}\mathbf{N}_{\mathbf{y}}\bar{v}-\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}\mathbf{P}_{\mathbf{y}}\bar{d}\end{aligned}\tag{5.37}$$

This gives the relationships between the optimal values of $\bar{u}$ and the disturbances coming into the plant (setpoints, upstream outputs, costs, etc.). Now examine the final values of these terms as the information system dynamics convergence to steady state. These dynamics (5.30) collapse to the following steady-state relationships:

$$\begin{aligned}&\bar{u}\rightarrow u^{QP,NC}\\&\bar{v}\rightarrow\left(I-\mathbf{N}_{\mathbf{z}}\Phi\right)^{-1}\mathbf{M}_{\mathbf{z}}\bar{u}+\left(I-\mathbf{N}_{\mathbf{z}}\Phi\right)^{-1}\mathbf{P}_{\mathbf{z}}\bar{d}=\mathbf{W}\bar{u}+\mathbf{X}\bar{d}\\&\psi\rightarrow 2\left(I-\Phi^{T}\mathbf{N}_{\mathbf{z}}{}^{T}\right)^{-1}\Phi^{T}\mathbf{N}_{\mathbf{y}}{}^{T}\mathbf{Q}\left(\mathbf{M}_{\mathbf{y}}\bar{u}+\mathbf{N}_{\mathbf{y}}\bar{v}+\mathbf{P}_{\mathbf{y}}\bar{d}-\mathbf{r}\right)\end{aligned}\tag{5.38}$$

Substituting (5.38) into (5.37) above yields:

$$\begin{aligned}&\left(\mathbf{S}+\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}\mathbf{M}_{\mathbf{y}}\right)\bar{u}^{QP,NC}\\&\quad=\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}\left(r-\mathbf{P}_{\mathbf{y}}\bar{d}\right)-\mathbf{M}_{\mathbf{y}}{}^{T}\mathbf{Q}\mathbf{N}_{\mathbf{y}}\left(\mathbf{W}\bar{u}^{QP,NC}+\mathbf{X}\bar{d}\right)\\&\quad-\mathbf{M}_{\mathbf{z}}{}^{T}\left(I-\Phi^{T}\mathbf{N}_{\mathbf{z}}{}^{T}\right)^{-1}\Phi^{T}\mathbf{N}_{\mathbf{y}}{}^{T}\mathbf{Q}\begin{pmatrix}\mathbf{M}_{\mathbf{y}}\bar{u}^{QP,NC}-r+\mathbf{P}_{\mathbf{y}}\bar{d}\\+\mathbf{N}_{\mathbf{y}}\left(\mathbf{W}\bar{u}^{QP,NC}+\mathbf{X}\bar{d}\right)\end{pmatrix}\end{aligned}\tag{5.39}$$

Rearranging terms:

$$\begin{aligned}&\left(\mathbf{S}+\left(\mathbf{M}_{\mathbf{y}}+\mathbf{N}_{\mathbf{y}}\mathbf{W}\right)^{T}\mathbf{Q}\left(\mathbf{M}_{\mathbf{y}}+\mathbf{N}_{\mathbf{y}}\mathbf{W}\right)\right)\bar{u}^{QP,NC}\\&\quad=\left(\mathbf{M}_{\mathbf{y}}{}^{T}+\mathbf{W}^{T}\mathbf{N}_{\mathbf{y}}{}^{T}\right)\mathbf{Q}\left(r-\left(\mathbf{P}_{\mathbf{y}}{}^{T}+\mathbf{X}^{T}\mathbf{N}_{\mathbf{y}}{}^{T}\right)\bar{d}\right)\end{aligned}\tag{5.40}$$

Comparison of (5.40) to (5.35) shows that the relationship between the optimal $\bar{u}$ and the setpoint *r* and disturbance $\bar{d}$ for the NC-OPT case converges to the same as the centralized case. Note that the algorithm is able to do this even though the local controllers are solving different cost functions.

Soft Uncoupled Input Constraints

If a soft, uncoupled constraint is applied to each optimizer, then the local cost functions will have an extra term $g_i(\bar{u}_i)$ which is a nonlinear, differentiable function of $\bar{u}_i$:

$$J_i=\bar{e}_i^{T}Q_i\bar{e}_i+\bar{u}_i^{T}S_i\bar{u}_i+\psi_i^{T}\bar{z}_i+g_i(\bar{u}_i)\tag{5.41}$$

Define a stacked vector of these scalar functions thus:

$$\mathrm{g}(\bar{u})=\begin{bmatrix}g_1(\bar{u}_1) & \cdots & g_p(\bar{u}_p)\end{bmatrix}^{T}\tag{5.42}$$

Applying the same treatment as earlier results in a slightly different version of (5.35):

$$\begin{aligned}&\left(\mathbf{S}+\left(\mathbf{M}_{\mathbf{y}}+\mathbf{N}_{\mathbf{y}}\mathbf{W}\right)^{T}\mathbf{Q}\left(\mathbf{M}_{\mathbf{y}}+\mathbf{N}_{\mathbf{y}}\mathbf{W}\right)\right)\bar{u}^{OPT}+{\partial \mathrm{g}}/{\partial\bar{u}}\\&\quad=\left(\mathbf{M}_{\mathbf{y}}{}^{T}+\mathbf{W}^{T}\mathbf{N}_{\mathbf{y}}{}^{T}\right)\mathbf{Q}\left(r-\left(\mathbf{P}_{\mathbf{y}}{}^{T}+\mathbf{X}^{T}\mathbf{N}_{\mathbf{y}}{}^{T}\right)\bar{d}\right)\end{aligned}\tag{5.43}$$

Since the constraints on $\bar{u}$ *u* are uncoupled, $\partial g_i/\partial\bar{u}_{j\neq i}=0$ and the derivation for the NC-OPT case again results in the same cost as the centralized case.

CHAPTER VI

NC-OPT SIMULATION

The previous chapter presented an algorithm for networked systems that calculated a centralized optimum via iteration between local optimizers. In this chapter the NC-OPT algorithm is applied to a model of a 9-zone building with an HVAC system, and the steps to designing and implementation of the method are discussed. The chapter is organized as follows. First, the building is presented, including a discussion of modeling software. A division of the system into a network of subsystems is proposed, and the details of extracting linear models from the nonlinear system are presented. The controller design is detailed. Finally, the results of a series of simulations over a day in July are presented. Of particular note is that the NC-OPT algorithm is capable of selecting setpoints for not only zone temperatures, but also for air supply and chilled water temperature reset, and this capability is highlighted as well.

Modeling a Building Using EnergyPlus

The building model for use in these simulations was created in a program available from the U.S. Department of Energy called EnergyPlus [116]. This program allows users to model a building, complete with an HVAC system, and thoroughly analyze its behavior and characteristics for a wide range of situations. Since EnergyPlus is more of a software engine and does not feature a user-friendly “front end”, other programs must be used in conjunction with it for efficient work flow. The freeware

program SketchUp allows the user to model the building in 3-D, including fenestrations [117]. The software suite OpenStudio links SketchUp and EnergyPlus to each other, allowing for an easier workflow in setting up the building [118].

In order to create and run an NC-OPT controller with the building, more software packages are required. The simulation software Simulink, part of the MatLab suite, provides the base for programming the NC-OPT controller. A Simulink S-function was coded in the programming language C to generate the NC-OPT calculations, given the plant outputs and disturbances, as well as the user-desired setpoints. In order to interface the code with the model, the Building Controls Virtual Test Bed (BCVTB) software engine was used; this was developed at the Lawrence Berkeley National Laboratory in Berkeley, CA and is freely available for download [119]. This engine is also not especially user friendly, but the MLEP software from the University of Pennsylvania provides a front end that allows a Simulink block representing the EnergyPlus model to be dropped into a Simulink block diagram [120]. Figure 6.1 shows the workflow of using the collection of software to carry out the simulations in this chapter.

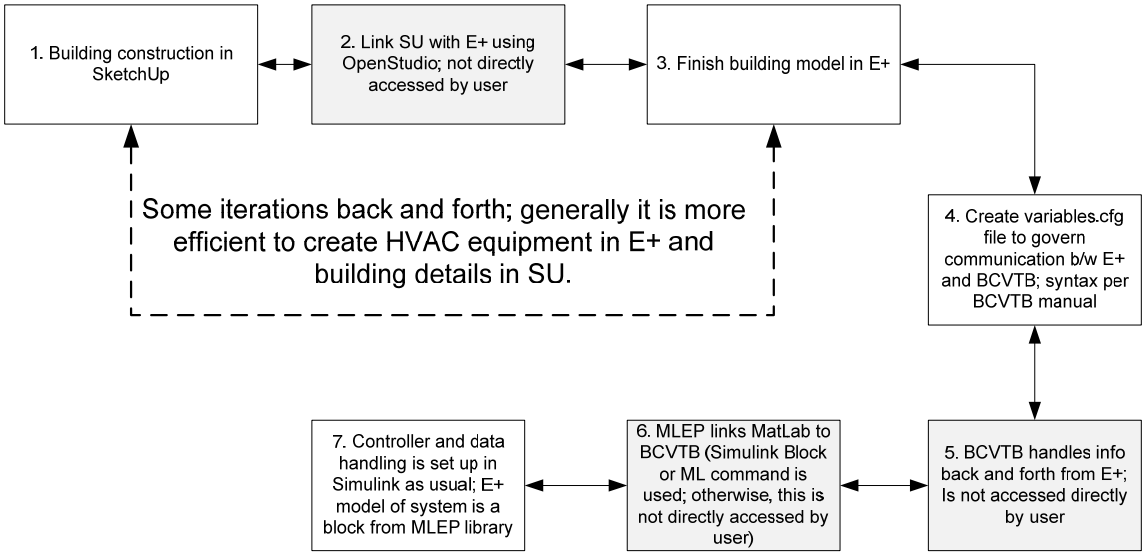


Figure 6.1: Workflow for setting up simulations

The building used in the simulations has nine zones, each corresponding to a different room; views of the building in SketchUp are shown in Figure 6.2. This building features a ring of offices around a central atrium, with two of the offices joined by a small hallway. There is a second-floor room over one side of the building.

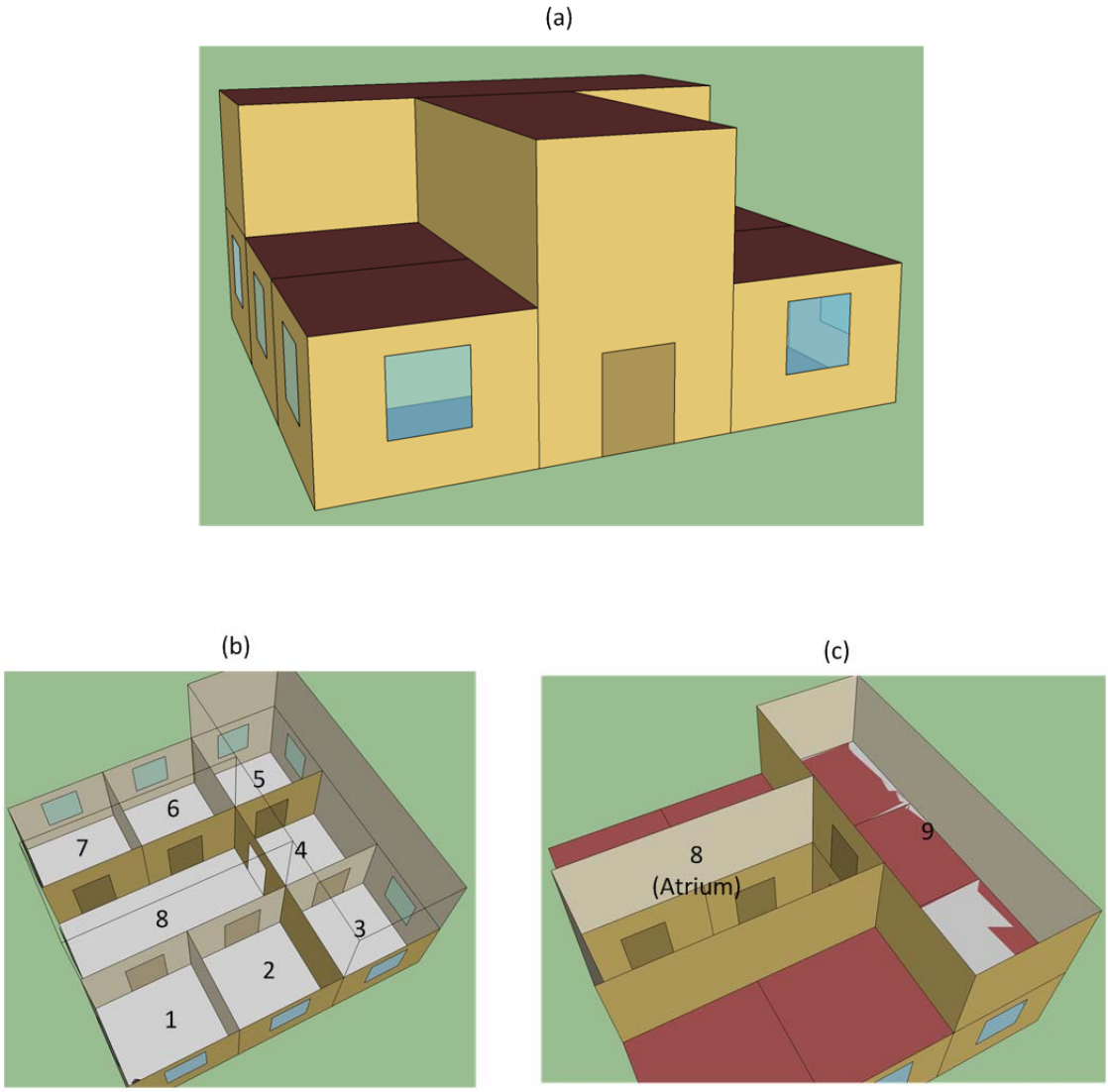


Figure 6.2: Views of simulation building. (a) Front view, (b) First floor (Zones 1-8), (c) Second floor (Zones 8 and 9). Note that Zone 8 is an atrium that spans both floors. Zone 4 is a small hallway between the offices in Zones 3 and 5.

The HVAC system is modeled using the HVAC System Templates capability in EnergyPlus. Rather than define each and every node and component in the HVAC system, the templates allow the modeler to quickly assemble a basic system for simulation. While this places some limitations on the complexity of the system that is used, the templates are sufficient for the presentation in this dissertation. The HVAC system for the simulation building features a 20 kilowatt electric water chiller that provides chilled water via a loop to a variable air volume (VAV) ventilation system. The temperature of this chilled water is regulated to a user-defined setpoint fed into the chiller model. The VAV system uses the chilled water to cool air returning from the zones; a variable speed fan keeps a constant pressure head in the ducting that supplies the zones. Additionally, the VAV system uses a chilled water coil to regulate the air supply temperature to a setpoint. Each zone has its own VAV terminal, which has a damper that opens and closes to regulate the temperature in each room according to setpoints defined by the occupant; thus, as the dampers open, the VAV system fan must spin faster, consuming more power. In the simulation, the building is in Tampa, Florida, and the simulations are carried out during the first week of July. Figure 6.3 shows a basic schematic of the control signals.

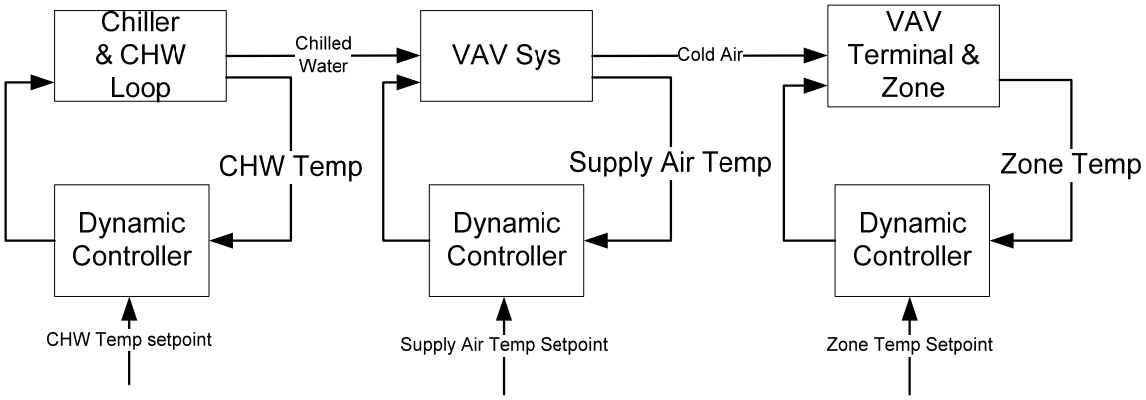


Figure 6.3: HVAC system components

Figure 6.4 shows how the subsystems are broken out into a network topology. Each of the zones is a subsystem. Since each zone's damper affects the fan power consumption of the VAV system, the damper positions are treated as disturbances to that system. Additionally, due to heat transfer between zones, the zone temperatures are also disturbances, but only between immediate neighbors. The air supply temperature is treated as a disturbance to the zones, since a higher air supply temperature may require more air flow to achieve the desired cooling. This temperature is also a disturbance to the chiller. The chilled water temperature is a disturbance to the VAV system. Exogenous, uncontrolled disturbances include the heat load in each zone as well as the outside air temperature.

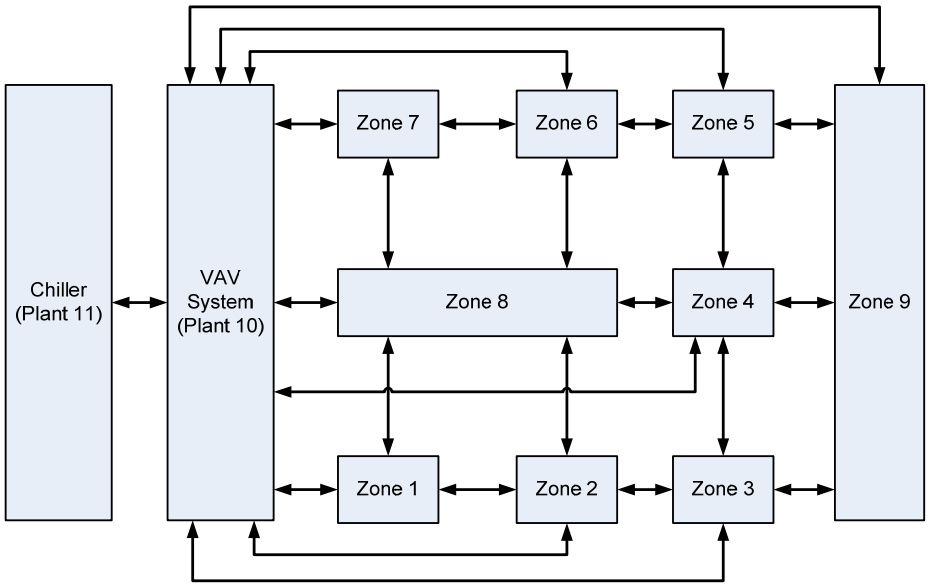


Figure 6.4: Network topology for the NC-OPT architecture as applied to the simulation system.

Subsystem Models

As mentioned in the previous section, each zone is a subsystem in this controller design. The subsystem accepts a zone temperature setpoint as its input $\overline{u}_i$ from the local optimizer. Its tracking outputs are the zone temperature; the optimizer modulates the setpoint requested by the user or occupant and balances that desired temperature with the energy consumption required. The energy consumption information is communicated to the local controller via the NC-OPT communication, since the local zone optimizers by design have no knowledge of system energy consumption. The outputs $\overline{z}_i$ that act as disturbances to its downstream neighbors are the zone temperature (to the other rooms) and the predicted steady-state VAV damper position (to the VAV system). The disturbances from the upstream plants are the air supply temperature and the zone

temperatures of the neighboring zones. The exogenous disturbances are the outside air temperature and the sensible zone cooling, which represents heat load in the zone. Figure 6.5 is a typical zone block diagram.

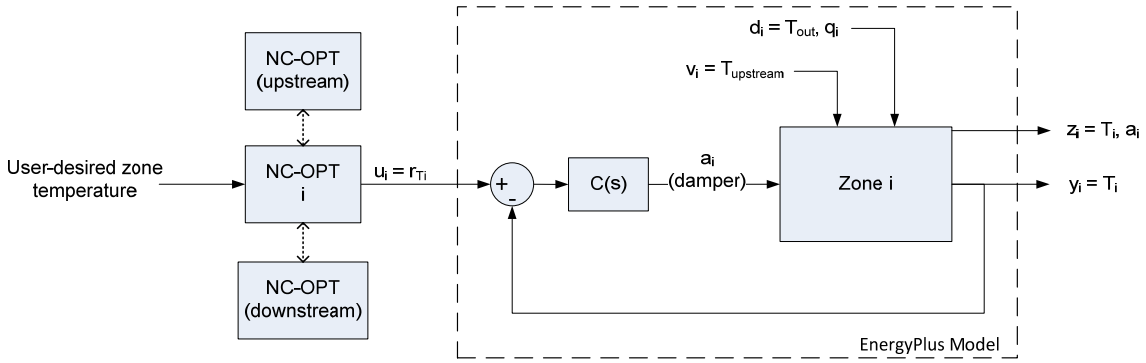


Figure 6.5: Block diagram for typical zone

The VAV system is also a subsystem. In this case, the input is an air supply temperature setpoint from the local optimizer; the optimizer will balance power consumption with tracking the design setpoint (air supply temperature reset). Its tracking outputs are the air supply temperature and the predicted power consumption of the fan. In order to minimize fan power consumption, the setpoint for power will be set to zero. The outputs that act as downstream disturbances are the air supply temperature (to the zones and to the chiller). Its exogenous disturbance is the outside air temperature. The block diagram is shown in Figure 6.6.

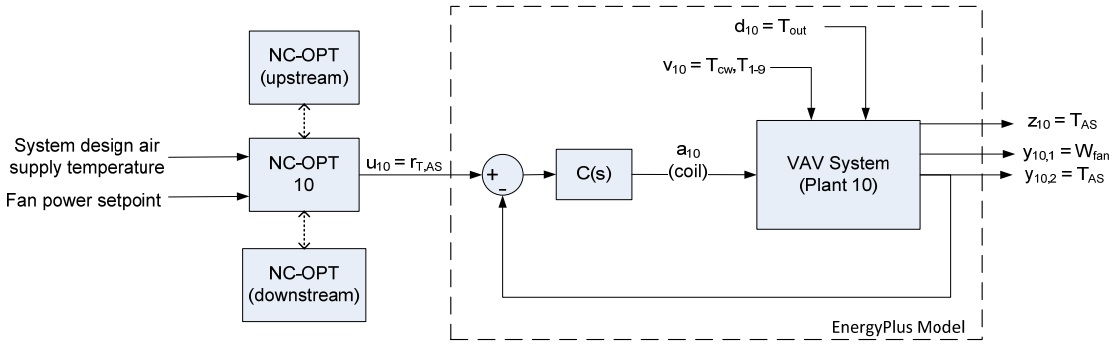


Figure 6.6: VAV subsystem

The last subsystem is the chiller. Its outputs are chilled water temperature and power consumption; as with the VAV subsystem, a setpoint of 0 for power will seek to minimize power consumption for the chiller. It accepts a chilled water temperature setpoint; the local optimizer is thus calculating a chilled water temperature reset. The subsystem's exogenous disturbance is the outside air temperature. Its upstream disturbance is the air supply temperature, and the chilled water temperature is also a disturbance to the VAV system. The block diagram of the subsystem shown in Figure 6.7.

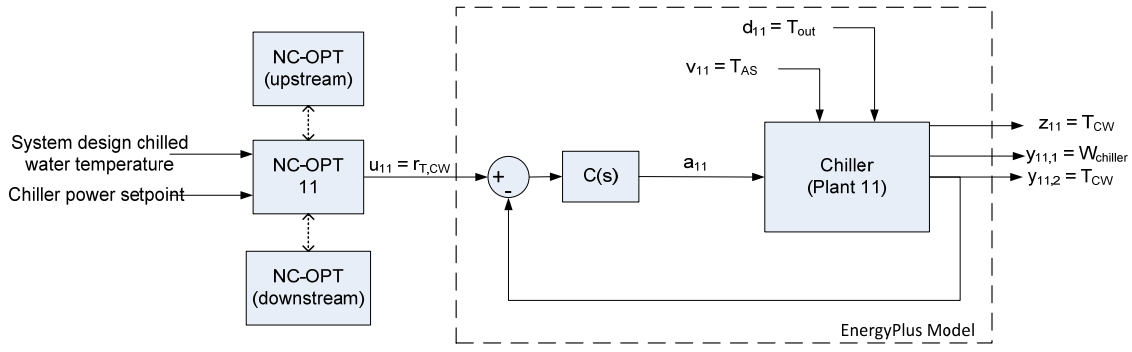


Figure 6.7: Chiller subsystem

Once the network topology and the local subsystems were selected, the steady state models were developed. A baseline simulation was run first. The model was run for a simulation that lasted 8 days of simulation time, from July 1 to July 8. The setpoints fed to the model for this baseline test (and the corresponding outputs) are treated as the system origin. The setpoints used for the baseline were 22°C for the zone setpoints; 10°C for the air supply temperature, and 6°C for the chilled water supply. To obtain the gains that the NC-OPT matrices consist of, a series of simulations were run wherein the setpoints were changed one at a time. For example, a simulation was run with a setpoint of 23°C sent to zone 1; all other setpoints were kept at the origin. The damper position from the baseline test was subtracted from the damper position during the new test; a least squares fit was used to find the gain from the change in temperature to the change in damper position. This process was repeated for all of the inputs and ouputs. To find the gain from the outside temperature to the various outputs, the average temperature was used for the baseline, since the same weather profile was used for all simulations. Additionally, extra simulations were run with different interior heat loads, so that the gain from sensible zone cooling to the various outputs could be found. The models are given in the following equations.

Once the models are found, weights must be selected. Apart from the error and actuator weights in **Q** and **S**, the two main parameters that the designer must tune are the convergence parameter *w* and the number of times the optimizers communicate per iteration cycle (c_{max}). The time step between optimizations must also be selected. The next section explores the effects of the different parameters.

Subsystem 1 (Zone 1):

$$[\bar{y}_1] = [T_1] = [1]\bar{u}_1 + \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} T_2 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix}\begin{bmatrix} T_{out} \\ q_1 \end{bmatrix}$$

$$\begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \\ \bar{z}_3 \end{bmatrix} = \begin{bmatrix} T_1 \\ T_1 \\ a_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -0.043 \end{bmatrix}\bar{u}_1 + (10^{-3})\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 4.8 & 6.1 & 8.9 \end{bmatrix}\begin{bmatrix} T_2 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0259 & 0.435 \end{bmatrix}\begin{bmatrix} T_{out} \\ q_1 \end{bmatrix} \tag{6.1}$$

Subsystem 2 (Zone 2):

$$[\bar{y}_2] = [T_2] = [1]\bar{u}_2 + \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} T_1 \\ T_3 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix}\begin{bmatrix} T_{out} \\ q_2 \end{bmatrix}$$

$$\begin{bmatrix} \bar{z}_4 \\ \bar{z}_5 \\ \bar{z}_6 \\ \bar{z}_7 \end{bmatrix} = \begin{bmatrix} T_2 \\ T_2 \\ T_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -0.038 \end{bmatrix}\bar{u}_2 + (10^{-3})\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 4.8 & 4.5 & 5.6 & 6.0 \end{bmatrix}\begin{bmatrix} T_1 \\ T_3 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0143 & 0.429 \end{bmatrix}\begin{bmatrix} T_{out} \\ q_2 \end{bmatrix} \tag{6.2}$$

Subsystem 3 (Zone 3):

$$[\bar{y}_3]=[T_3]=[1]\bar{u}_3+\begin{bmatrix}0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}T_2\\T_4\\T_9\\T_{AS}\end{bmatrix}+\begin{bmatrix}0 & 0\end{bmatrix}\begin{bmatrix}T_{out}\\q_3\end{bmatrix}$$

$$\begin{bmatrix}\bar{z}_8\\\bar{z}_9\\\bar{z}_{10}\\\bar{z}_{11}\end{bmatrix}=\begin{bmatrix}T_3\\T_3\\T_3\\a_3\end{bmatrix}=\begin{bmatrix}1\\1\\1\\-0.036\end{bmatrix}\bar{u}_3+(10^{-3})\begin{bmatrix}0 & 0 & 0 & 0\\0 & 0 & 0 & 0\\4.5 & 4.7 & 5.1 & 5.9\end{bmatrix}\begin{bmatrix}T_2\\T_4\\T_9\\T_{AS}\end{bmatrix} \qquad (6.3)$$

$$+\begin{bmatrix}0 & 0\\0 & 0\\0.0152 & 0.423\end{bmatrix}\begin{bmatrix}T_{out}\\q_3\end{bmatrix}$$

Subsystem 4 (Zone 4):

$$[\bar{y}_4]=[T_4]=[1]\bar{u}_4+\begin{bmatrix}0 & 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}T_3\\T_5\\T_8\\T_9\\T_{AS}\end{bmatrix}+\begin{bmatrix}0 & 0\end{bmatrix}\begin{bmatrix}T_{out}\\q_4\end{bmatrix}$$

$$\begin{bmatrix}\bar{z}_{12}\\\bar{z}_{13}\\\bar{z}_{14}\\\bar{z}_{15}\\\bar{z}_{16}\end{bmatrix}=\begin{bmatrix}T_4\\T_4\\T_4\\T_4\\a_4\end{bmatrix}=\begin{bmatrix}1\\1\\1\\1\\-0.032\end{bmatrix}\bar{u}_4+(10^{-3})\begin{bmatrix}0 & 0 & 0 & 0\\0 & 0 & 0 & 0\\5.8 & 5.8 & 5.6 & 2.7\end{bmatrix}\begin{bmatrix}T_3\\T_5\\T_8\\T_9\\T_{AS}\end{bmatrix} \qquad (6.4)$$

$$+\begin{bmatrix}0 & 0\\0 & 0\\0.003 & 0.511\end{bmatrix}\begin{bmatrix}T_{out}\\q_4\end{bmatrix}$$

Subsystem 5 (Zone 5):

$$[\bar{y}_5] = [T_5] = [1]\bar{u}_5 + \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_4 \\ T_6 \\ T_9 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_5 \end{bmatrix}$$

$$\begin{bmatrix} \bar{z}_{17} \\ \bar{z}_{18} \\ \bar{z}_{19} \\ \bar{z}_{20} \end{bmatrix} = \begin{bmatrix} T_5 \\ T_5 \\ T_5 \\ a_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -0.036 \end{bmatrix} \bar{u}_5 + (10^{-3}) \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 4.7 & 4.4 & 5.1 & 6.0 \end{bmatrix} \begin{bmatrix} T_4 \\ T_6 \\ T_9 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0148 & 0.422 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_5 \end{bmatrix} \tag{6.53}$$

Subsystem 6 (Zone 6):

$$[\bar{y}_6] = [T_6] = [1]\bar{u}_6 + \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_5 \\ T_7 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_6 \end{bmatrix}$$

$$\begin{bmatrix} \bar{z}_{21} \\ \bar{z}_{22} \\ \bar{z}_{23} \\ \bar{z}_{24} \end{bmatrix} = \begin{bmatrix} T_6 \\ T_6 \\ T_6 \\ a_6 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -0.035 \end{bmatrix} \bar{u}_6 + (10^{-3}) \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 4.4 & 4.7 & 5.4 & 4.7 \end{bmatrix} \begin{bmatrix} T_5 \\ T_7 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0098 & 0.427 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_6 \end{bmatrix} \tag{6.6}$$

Subsystem 7 (Zone 7):

$$[\bar{y}_7] = [T_7] = [1]\bar{u}_7 + \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_6 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_7 \end{bmatrix}$$

$$\begin{bmatrix} \bar{z}_{25} \\ \bar{z}_{26} \\ \bar{z}_{27} \end{bmatrix} = \begin{bmatrix} T_7 \\ T_7 \\ a_7 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -0.043 \end{bmatrix} \bar{u}_7 + (10^{-3}) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 4.6 & 6.1 & 9.0 \end{bmatrix} \begin{bmatrix} T_6 \\ T_8 \\ T_{AS} \end{bmatrix} \tag{6.7}$$

$$+ \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0256 & 0.434 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_7 \end{bmatrix}$$

Subsystem 9 (Zone 9):

$$[\bar{y}_9] = [T_9] = [1]\bar{u}_9 + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_3 \\ T_4 \\ T_5 \\ T_8 \\ T_{AS} \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_9 \end{bmatrix}$$

$$\begin{bmatrix} \bar{z}_{35} \\ \bar{z}_{36} \\ \bar{z}_{37} \\ \bar{z}_{38} \\ \bar{z}_{39} \end{bmatrix} = \begin{bmatrix} T_9 \\ T_9 \\ T_9 \\ T_9 \\ a_9 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -0.078 \end{bmatrix} \bar{u}_9 + (10^{-3}) \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 4.6 & 4.1 & 4.7 & 6.4 & 15.8 \end{bmatrix} \begin{bmatrix} T_3 \\ T_4 \\ T_5 \\ T_8 \\ T_{AS} \end{bmatrix} \tag{6.8}$$

$$+ \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0293 & 0.422 \end{bmatrix} \begin{bmatrix} T_{out} \\ q_9 \end{bmatrix}$$

Subsystem 8 (Zone 8):

$$\left[\overline{y}_8\right]=\left[T_8\right]=\left[1\right]\overline{u}_8+\begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}T_1\\T_2\\T_4\\T_6\\T_7\\T_9\\T_{AS}\end{bmatrix}+\begin{bmatrix}0 & 0\end{bmatrix}\begin{bmatrix}T_{out}\\q_8\end{bmatrix}$$

$$\begin{bmatrix}\overline{z}_{28}\\\overline{z}_{29}\\\overline{z}_{30}\\\overline{z}_{31}\\\overline{z}_{32}\\\overline{z}_{33}\\\overline{z}_{34}\end{bmatrix}=\begin{bmatrix}T_8\\T_8\\T_8\\T_8\\T_8\\T_8\\a_8\end{bmatrix}=\begin{bmatrix}1\\1\\1\\1\\1\\1\\-0.085\end{bmatrix}\overline{u}_8$$

$$+(10^{-3})\begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 0 & 0\\6.1 & 5.6 & 3.3 & 5.3 & 6.2 & 7.0 & 13.3\end{bmatrix}\begin{bmatrix}T_1\\T_2\\T_4\\T_6\\T_7\\T_9\\T_{AS}\end{bmatrix} \qquad (6.9)$$

$$+\begin{bmatrix}0 & 0\\0 & 0\\0.0252 & 0.440\end{bmatrix}\begin{bmatrix}T_{out}\\q_1\end{bmatrix}$$

Subsystem 10 (VAV system):

$$\begin{bmatrix} \overline{y}_{10} \\ \overline{y}_{11} \end{bmatrix} = \begin{bmatrix} T_{AS} \\ \dot{W}_{fan} \end{bmatrix} = \begin{bmatrix} 1 \\ 0.0021 \end{bmatrix} \overline{u}_{10} + \begin{bmatrix} 0 & \cdots & 0 \\ & Nz_{10,2} & \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_9 \\ T_{CW} \end{bmatrix} + \begin{bmatrix} 0 \\ 0.0032 \end{bmatrix} T_{out}$$

$$\begin{bmatrix} \overline{z}_{40} \\ \vdots \\ \overline{z}_{49} \end{bmatrix} = \begin{bmatrix} T_{AS} \\ \vdots \\ T_{AS} \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \overline{u}_{10} + \begin{bmatrix} 0 & \cdots & 0 \\ & \vdots & \\ 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_9 \\ T_{CW} \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} T_{out} \tag{6.10}$$

$$Nz_{10,2} = 10^{-2}[2.33 \quad 1.77 \quad 1.36 \quad 0.70 \quad 1.35 \quad 1.68 \quad 2.31 \quad 1.71 \quad 2.17 \quad 0.43]$$

Subsystem 11 (Chiller):

$$\begin{bmatrix} \overline{y}_{12} \\ \overline{y}_{13} \end{bmatrix} = \begin{bmatrix} T_{CW} \\ \dot{W}_{ch} \end{bmatrix} = \begin{bmatrix} 1 \\ -0.0284 \end{bmatrix} \overline{u}_{11} + \begin{bmatrix} 0 \\ 0.0002 \end{bmatrix} [T_{AS}] + \begin{bmatrix} 0 \\ 0.07 \end{bmatrix} T_{out} \tag{6.11}$$

$$[\overline{z}_{50}] = [T_{CW}] = [1]\overline{u}_{11} + [0][T_{AS}] + [0]T_{out}$$

Simulations

NC-OPT Iterations

The first parameter to be examined is the convergence weight *w*. This term acts to help stabilize the information dynamics as the local optimizers find a solution. The presence of this term results in a convex combination of the previous time step and the previous time step's calculated input. This style of iteration is referred to as a Gauss-Seidel algorithm [121]. Figure 6.8 shows the convergence of three different convergence weights to the centralized optimum solution. The iterations begin with

each setpoint at the system origin detailed earlier; i.e., no chilled water or air supply temperature resets, and no modulation of the zone temperatures. The local optimizers converge to the centralized solution with relatively little iteration, depending upon the convergence term *w*, which is a design parameter in the NC-OPT algorithm. For this particular system, a small convergence term (and hence a fast convergence) is useable, which allows for a stable convergence in less than 5 iterations. For all of the following simulations, a weight of 1.0 is placed on the chilled water and air supply temperature errors, with a weight of 100 placed on the power consumption by the VAV fan and the water chiller. The weights upon the zone errors are varied in the simulations; for the iterations shown in Figure 6.8 and 6.9 a weight of 0.1 is used.

Figure 6.9 displays how one of the zones (Zone 2) iterates to a new zone temperature as the cost information from its downstream neighbors is communicated to the optimizer. The components of the vector ψ_2 are shown in 6.9(a); these are costs calculated by each of Zone 2's neighbors using the function for γ and passes to the Zone 2 optimizer. The costs coming from the different zones are positive, since a warmer zone 2 temperature means that the zones would require more cooling to match their setpoints. Also, the cost values coming from the VAV system are negative, since a higher zone temperature would require less cooling, and the VAV fan would consume less power.

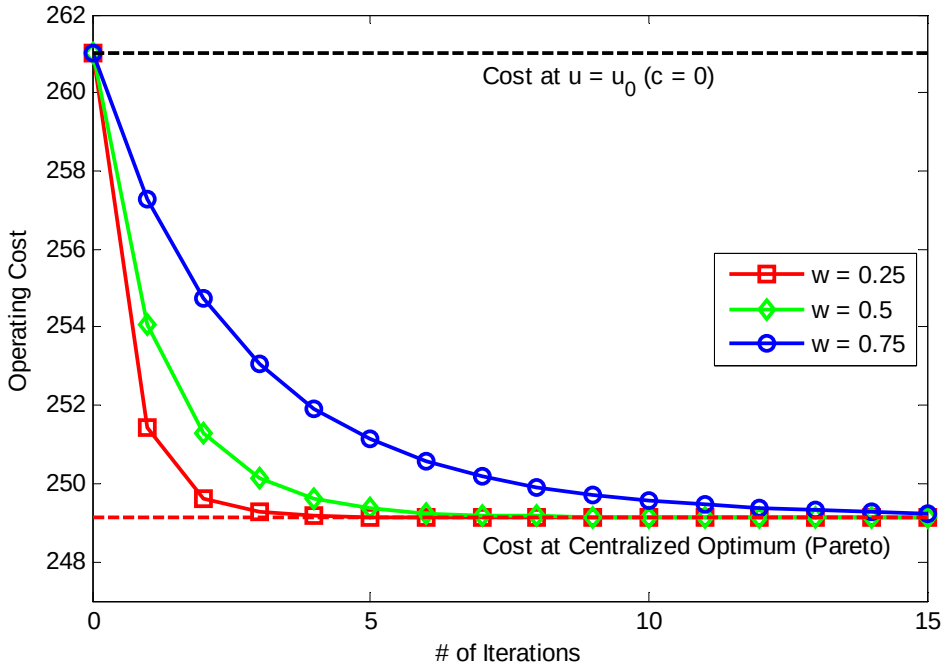


Figure 6.8: Iterations from user-desired setpoints to centralized optimum.

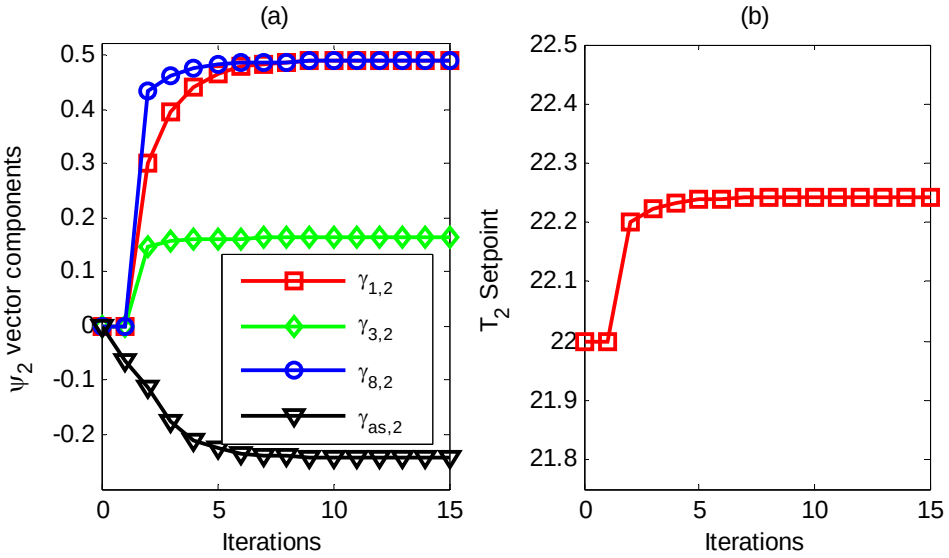


Figure 6.9: Zone 2 iterations. (a) Penalties on output $\bar{z}_2$ calculated by zones 1,3, and 8, and the VAV system. (b) Zone 2 temperature setpoint over iterations

Temperature Regulation in Dynamic Simulations

Figure 6.10 shows the temperatures for three of the nine zones for three different simulations (Zones 1, 4, and 6). These are chosen as representative behavior for all of the zones. The environmental conditions of the simulations are identical in each case, and are the same weekday in July. Each of the simulations was run with a different weight on the zone temperature error. A higher error implies that less deviation from the user-desired temperature will be tolerated, but at the cost of higher energy consumption. All simulations presented herein use parameters of $w = 0.75$ and $c_{max} = 25$, and the optimizers recalculate every 30 minutes.

The temperature setpoints increase during the hottest part of the day, since more power will be consumed to cool the zones at that time. This information is communicated to the zones from the optimizers for the VAV and chiller. As expected, the higher error weights result in closer adherence to the user-desired setpoints, although some zones react more strongly to changes in error weight than others. While most of the zones increase their temperatures as the weight goes down, Zone 4 behaves in the opposite manner: as its weight decreases, the zone temperature goes down instead of up as the others do. This is due to the geometry of the building—by reducing its temperature, zone 4 acts as a heat sink, reducing the cooling required by the other zones. The behavior of the VAV system and chiller optimizers is shown in Figure 6.11. Only the results for zone error weight of 0.1 are shown, since the other weights yielded similar results (within a few percent). The air supply temperature is reset by a very small amount (approximate 0.25 °C), but the chilled water temperature is modulated

significantly—as the day becomes warmer, the optimizer goes up to the limit of 10°C. Thus the NC-OPT architecture is not only choosing room temperatures, but also the temperatures of the chilled water and air supply.

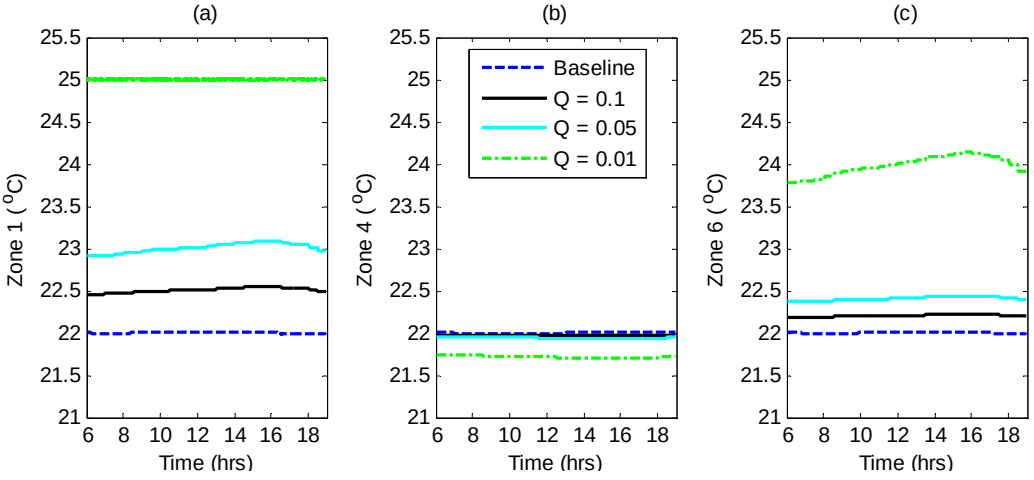


Figure 6.10: Temperatures for different weights for zones (a) 1, (b) 4, and (c) 6.

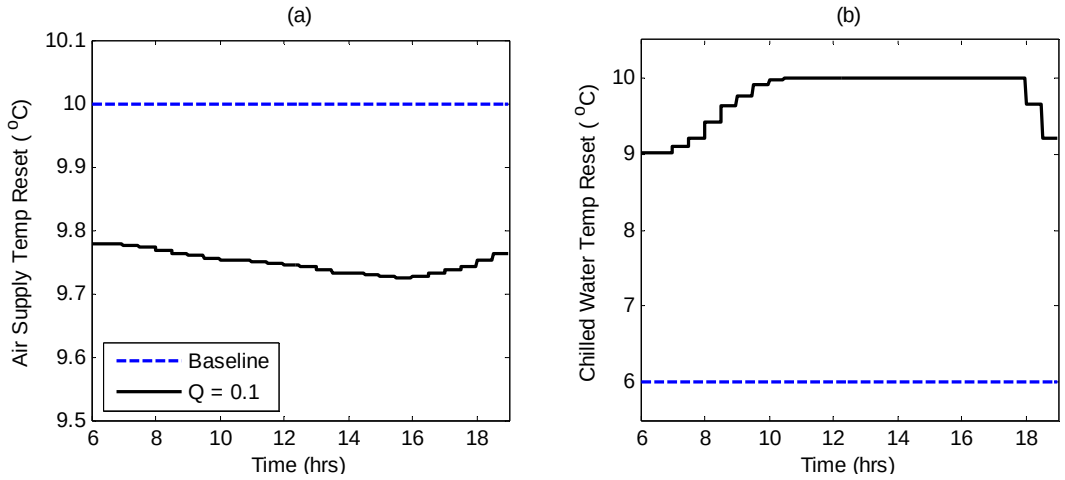


Figure 6.11: Temperatures for (a) Air supply and (b) Chilled water.

Energy Savings

The next two figures show the improvement in overall energy consumption using the NC-OPT algorithm as structured. Both the fan power and chiller power consumption are shown in Figure 6.12 for the simulations presented. As the zone error weight decreases, the chiller power consumption also decreases from the baseline case; however, the fan power actually increases for all but the most extreme case, i.e., a weight of 0.01. Figure 6.13, however, shows that the consumption for the entire system does indeed decrease as zone error weight decreases; for some cases, the fan works harder to deliver the cooling but the increase is more than offset by the decrease in chiller power consumption. Thus the components, through cooperation and iteration, reach a system-level optimum without a centralized controller.

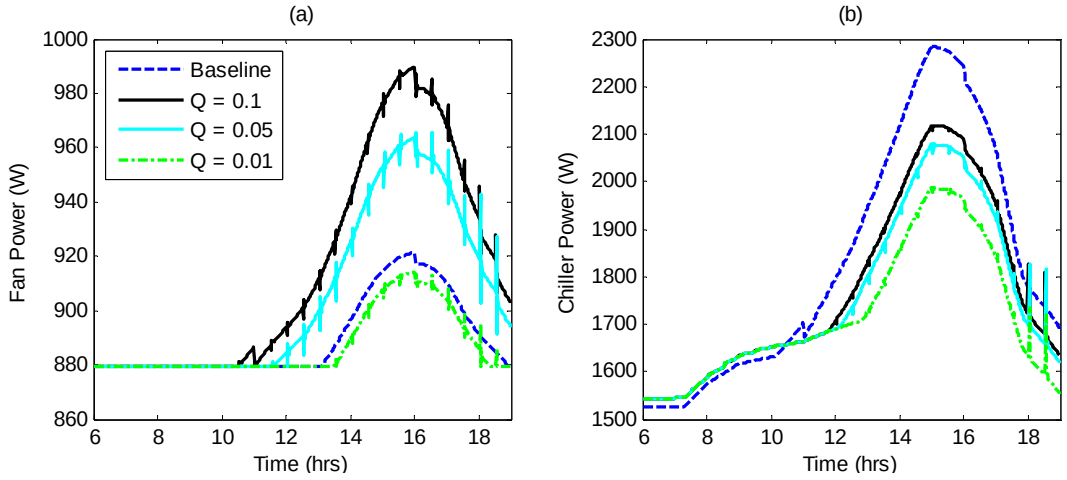


Figure 6.12: Power consumption for different weights for (a) the VAV system fan and (b) the chiller.

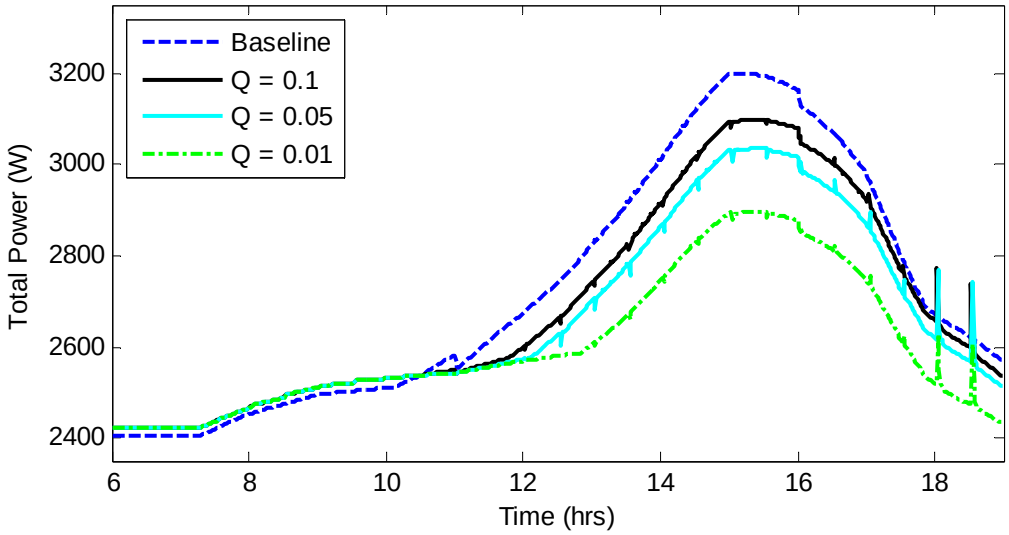


Figure 6.13: Total system power consumption for several different weights.

.

CHAPTER VII

NC-OPT DESIGN AND EXPERIMENTS

Another demonstration of the NC-OPT control approach is now given by application to the experimental system. A schematic of the system is given in Figure 7.1, labeled with the signals used throughout this chapter. The measurements, actuators, and disturbances are indicated on the figure and described in Table 7.1.

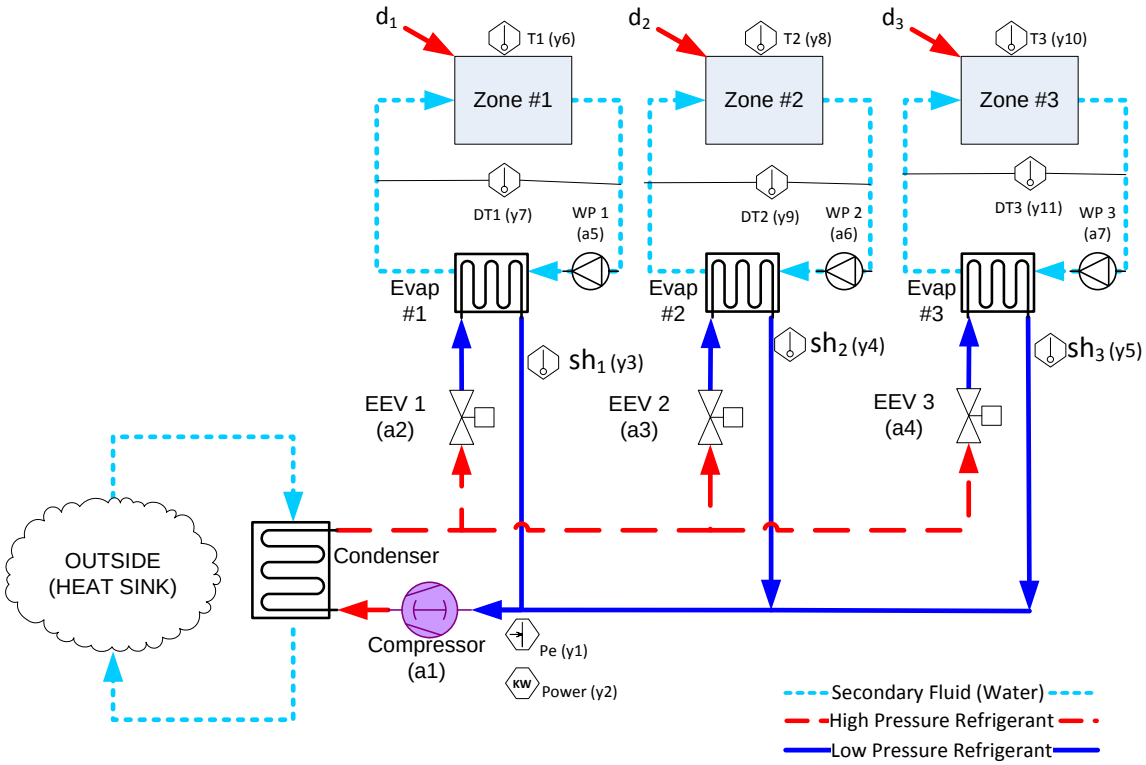


Figure 7.1: Experimental system schematic

Table 7.1: Signals in Figure 7.1

Actuators

Actuator Name	Description	Schematic Designation
Compressor	Compressor speed (RPM)	a_1
EEV 1, EEV 2, EEV 3	Electronic Expansion Valve (%)	a_2, a_3, a_4
WP 1, WP 2, WP 3	Evaporator Water Pump (%)	a_5, a_6, a_7

Sensors/Outputs

Output	Signal Description	NC-OPT Output	User Setpoint	NC-OPT Input
Pe	Suction pressure (kPa)	y_1	r_1	u_1
W	Compressor power draw (kW)	y_2	r_2	N/A
sh_1, sh_2, sh_3	Evaporator Superheats (°C)	y_3, y_4, y_5	r_3, r_4, r_5	u_2, u_3, u_4
T_1, T_2, T_3,	Zone Temperature (°C)	y_6, y_8, y_{10}	r_6, r_8, r_{10}	N/A
DT_1, DT_2, DT_3	Water Temperature Drop (°C)	y_7, y_9, y_{11}	r_7, r_9, r_{11}	u_5, u_6, u_7

Exogenous Disturbances

d_1, d_2, d_3	Heat load disturbance to Zone 1, 2, 3	d_1, d_2, d_3

Network Topology

The first step in applying the NC-OPT algorithm is dividing the networked system into individual subsystems. In this application, the subsystems are demarcated by the individual actuators, creating a network of single-input systems. Each of these systems has a single actuator that is dynamically controlled with a proportional-integral (PI) controller. Thus, as before, the $\overline{u}_i$ terms that the local optimizers are calculating are actually setpoints for these lower level controllers. Since the controllers are PI, a zero steady state error can be assumed from the controller reference setpoint to the regulated output. Therefore, many of the gains become zero. This simplifies the interconnection model greatly, since the controllers will reject disturbances from the upstream plants, assuming the actuators do not saturate. Figure 7.2 shows the network toplogy for the system as actually implemented; as a comparison, Figure 7.3 shows the interconnection for the system if a non-integrating control were used, such as a proportional controller or if the NC-OPT algorithm was calculating steady-state actuator positions.

In this control architecture, the compressor speed is used to control the suction pressure, the EEVs are used to control evaporator superheats, and the water pumps are used to control the water temperature drop, which is defined here as the temperature drop of the water as it passes through the evaporator. The specific models and local structure are detailed next.

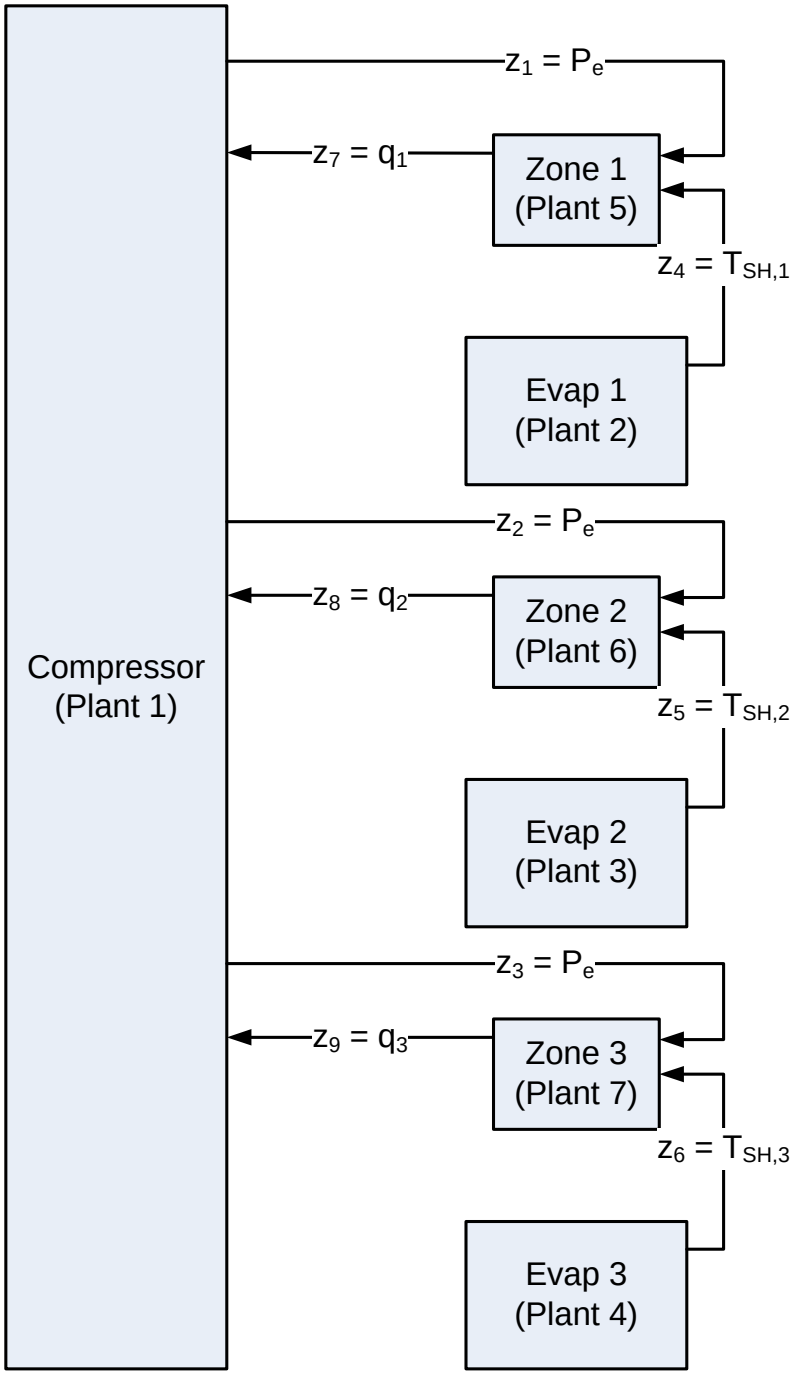


Figure 7.2: Network Topology of Experimental system, as implemented

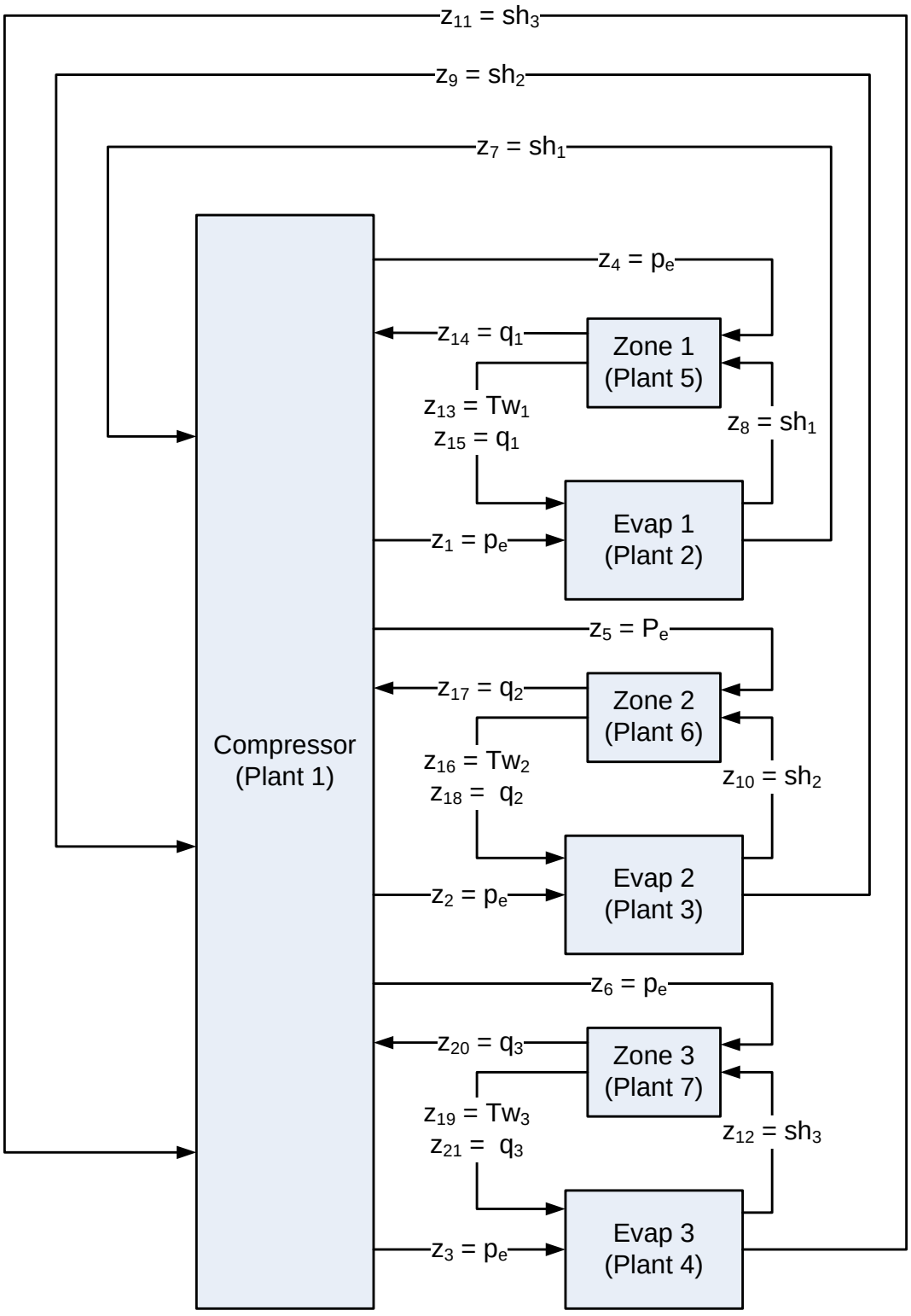


Figure 7.3: Network topology for system without integral control

Models

Compressor and condenser

The compressor in this system is a variable speed model, so an electronic controller specifies the rotational speed of the compressor in rotations per minute (RPM). The water pump that pushes water through the condenser only operates at a single speed, although in some systems this pump (or fan in an air-based system) has variable speed capacity. In these cases, an extra subsystem could be added to the architecture controlling this pump or fan.

The first subsystem (plant 1) is the combination of these two components. The pressure at the compressor inlet, referred to as the suction pressure, is strongly affected by the compressor speed, so the compressor is used to regulate the suction pressure to a setpoint. The compressor power consumption is also an output of this subsystem, although it is not directly controlled with a lower level controller. Since it is part of the $\bar{y}$ vector, the NC-OPT algorithm will take this consumption into account when calculating its inputs $\bar{u}_i$ to the system, subject to the error weight Q_i.

Since the compressor controller has an integrator, any disturbances to the suction pressure will be compensated for in the steady state model. Additionally, since the effect of changes in superheat is small due to the presence of a maximum in the effect of superheat on efficiency [122], the effect of superheat on power is ignored in this model. The block diagram for the dynamical subsystem is presented in Figure 7.4. The steady state model simplifies to:

$$\begin{bmatrix} \overline{y}_1 \\ \overline{y}_2 \end{bmatrix} = \begin{bmatrix} P_e \\ \dot{W} \end{bmatrix} = \begin{bmatrix} 1 \\ -0.0045 \end{bmatrix} u_1 + \begin{bmatrix} 0 & 0 & 0 \\ 0.296 & 0.296 & 0.296 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}$$

$$\begin{bmatrix} \overline{z}_1 \\ \overline{z}_2 \\ \overline{z}_3 \end{bmatrix} = \begin{bmatrix} P_e \\ P_e \\ P_e \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} u_1 + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (7.1)$$

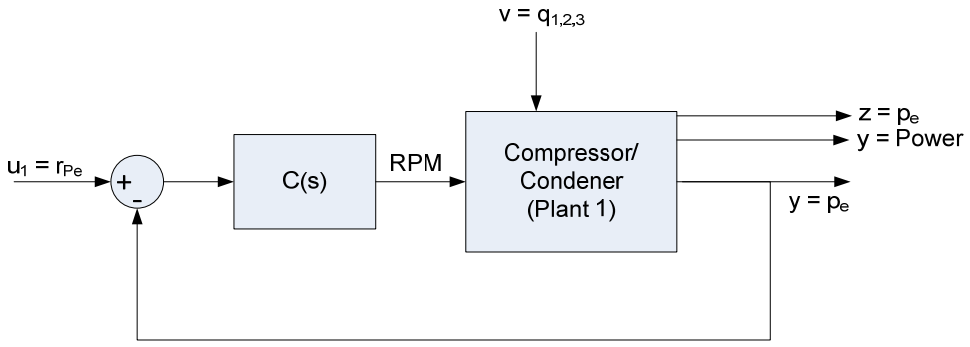


Figure 7.4: Subsystem 1 Block Diagram

Evaporators

The evaporators in a VCC system function by absorbing thermal energy from a zone into the refrigerant; this heat transfer process is referred to as cooling, and is measured in kilowatts (kW). Refrigerant is metered into the evaporator by the electronic expansion valve (EEV). Likewise, the variable speed water pump (WP) pumps water from the zone through the secondary passage of the evaporator and returns the chilled water back into the zone. (Note that in an air-based system this function would be performed by a fan blowing air across the evaporator coils.) For the purposes of this

application of the NC-OPT algorithm, these fluid flow control functions are split up by zone and by fluid.

Thus subsystems 2 through 4 are plants that have evaporator superheat as the output. An EEV with a PI controller regulates the superheat; the setpoint for the PI controller is the input for the subsystem. A block diagram is given in Figure 7.5. The EEVs at the evaporator inlets are used to regulate superheat to a setpoint; this removes all steady state disturbances and collapses the model to the following:

$$\begin{aligned} \overline{y}_{i+1} &= \left[sh_{i-1} \right] = [1]\overline{u}_i \\ \overline{z}_{i+2} &= [1]\overline{u}_i \\ i &= 2,3,4 \end{aligned} \tag{7.2}$$

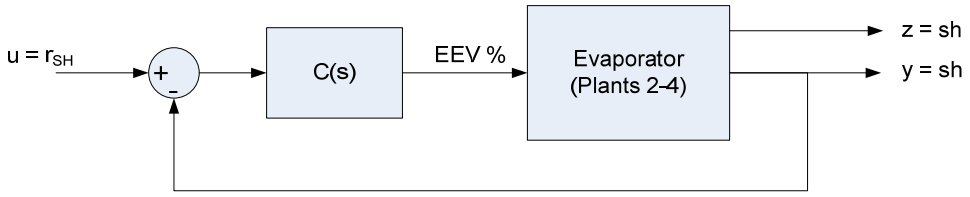


Figure 7.5: Typical block diagram, evaporators (subsystems 2-4).

Zones

Subsystems 5 through 7 are the zones themselves; these plants have the water pump speed (expressed as a percentage of full speed) as the input and water temperature drop and zone temperatures as the regulated outputs. Heat load is the disturbance for these plants, and is generated by an immersion heater placed in the water tank. The evaporator cooling is the signal fed downstream. Since these are steady state models,

the predicted temperature is unchanging; therefore, disturbance heat load (energy going into the zone) must equal the evaporator cooling capacity (energy going out of the zone). This means that the gain of the heat load disturbance to cooling capacity is 1. The same Kalman filter used in Chapter IV estimates the size of the heat load disturbance and feeds that information into the local optimizers. Figure 7.6 gives the block diagram for the subsystems. The signals are summarized in Tables 7.2 and 7.3. The steady state models used by the optimizers are:

$$\begin{bmatrix} \overline{y}_6 \\ \overline{y}_7 \end{bmatrix} = \begin{bmatrix} T_{w,1} \\ \Delta T_1 \end{bmatrix} = \begin{bmatrix} 1.13 \\ 1 \end{bmatrix} \overline{u}_5 + \begin{bmatrix} 1.09 & 0.07 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} sh_1 \\ p_e \end{bmatrix} + \begin{bmatrix} 4.08 \\ 0 \end{bmatrix} \overline{d}_1$$
$$\begin{bmatrix} \overline{z}_7 \end{bmatrix} = \begin{bmatrix} q_1 \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix} \overline{u}_5 + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} sh_1 \\ p_e \end{bmatrix} + \begin{bmatrix} 1 \end{bmatrix} \overline{d}_1 \tag{7.3}$$

$$\begin{bmatrix} \overline{y}_8 \\ \overline{y}_9 \end{bmatrix} = \begin{bmatrix} T_{w,2} \\ \Delta T_2 \end{bmatrix} = \begin{bmatrix} 0.80 \\ 1 \end{bmatrix} \overline{u}_6 + \begin{bmatrix} 0.998 & 0.079 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} sh_2 \\ p_e \end{bmatrix} + \begin{bmatrix} 4.08 \\ 0 \end{bmatrix} \overline{d}_2$$
$$\begin{bmatrix} \overline{z}_8 \end{bmatrix} = \begin{bmatrix} q_2 \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix} \overline{u}_6 + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} sh_2 \\ p_e \end{bmatrix} + \begin{bmatrix} 1 \end{bmatrix} \overline{d}_2 \tag{7.4}$$

$$\begin{bmatrix} \overline{y}_{10} \\ \overline{y}_{11} \end{bmatrix} = \begin{bmatrix} T_{w,3} \\ \Delta T_3 \end{bmatrix} = \begin{bmatrix} 1.21 \\ 1 \end{bmatrix} \overline{u}_7 + \begin{bmatrix} 1.41 & 0.065 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} sh_3 \\ p_e \end{bmatrix} + \begin{bmatrix} 4.08 \\ 0 \end{bmatrix} \overline{d}_3$$
$$\begin{bmatrix} \overline{z}_9 \end{bmatrix} = \begin{bmatrix} q_3 \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix} \overline{u}_7 + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} sh_3 \\ p_e \end{bmatrix} + \begin{bmatrix} 1 \end{bmatrix} \overline{d}_3 \tag{7.5}$$

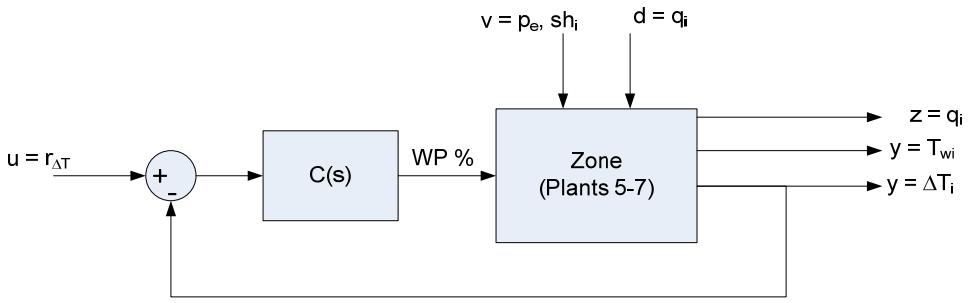


Figure 7.6: Typical block diagram, zones (subsystems 5-7).

Table 7.2: Output Signals

Regulated Output Signal	Plant #	y#/r #	Error weight Q_i:	Origin value (y_0)
Suction Pressure	1	1	0.01	410
Compressor Power	1	2	1000	0.4893
Evaporator 1 Superheat	2	3	0.25	8
Evaporator 2 Superheat	3	4	0.25	8
Evaporator 3 Superheat	4	5	0.25	8
Zone 1 Temperature	5	6	1.00	22.5
Zone 1 Water Temperature Drop	5	7	0.75	4
Zone 2 Temperature	6	8	1.00	21.5
Zone 2 Water Temperature Drop	6	9	0.75	4
Zone 3 Temperature	7	10	0.05	21.7
Zone 3 Water Temperature Drop	7	11	0.75	4

Table 7.3: Input Signals

Input Signal	Plant #	u #	Value weight S_i:	Origin value (u_0)	Actuator
Suction Pressure Setpoint	1	1	0.0	410	Compressor
Evap 1 Superheat Setpoint	2	2	0.0	8	EEV 1
Evap 2 Superheat Setpoint	3	3	0.0	8	EEV 2
Evap 3 Superheat Setpoint	4	4	0.0	8	EEV 3
Zone 1 Temp. Drop Setpoint	5	5	0.0	4	WP 1
Zone 2 Temp. Drop Setpoint	6	6	0.0	4	WP 2
Zone 3 Temp. Drop Setpoint	7	7	0.0	4	WP 3

Illustration of Convergence—Simulation

To illustrate the convergence behavior of the controllers, a simulated test case is shown in the following figures. The models and setpoints are fed into the algorithm and the controllers iterate from the origin of the linear system model. Figure 7.7 compares the value of the *centralized* cost function over the iterations to the Pareto minimum; again, the NC-OPT cost gets quite close within 25 iterations. The centralized cost for the decentralized optima is also indicated in the figure for comparison. Figure 7.8 shows the convergence of the setpoints to the centralized optimum. The abscissa on each graph is the iteration count; the various ordinates are the setpoints fed to the local controllers. While over 100 iterations are needed to converge to the centralized optimum, the deviations from the optimal setpoint are relatively very small after 25 to 50 iterations—typical uncertainty on a temperature measurement, for example, is 0.5°C.

These observations suggest that full convergence in one time step is probably not necessary for this application. This highlights that if the dynamics of the physical plants are slow relative to the sample time of the optimizers, then a low number of iterations will not necessarily impact the performance of the setpoint calculation. This concept is explored in the first data set in the following section.

Another important point is that the local NC-OPT optimizers are not solving the same cost function as a centralized controller, but still converge to the same optimum. This is demonstrated in Figure 7.9, for the same condition as the earlier two figures. The cost function solved by subsystem 1 (compressor) after the iterations have converged is superimposed over the centralized cost function.

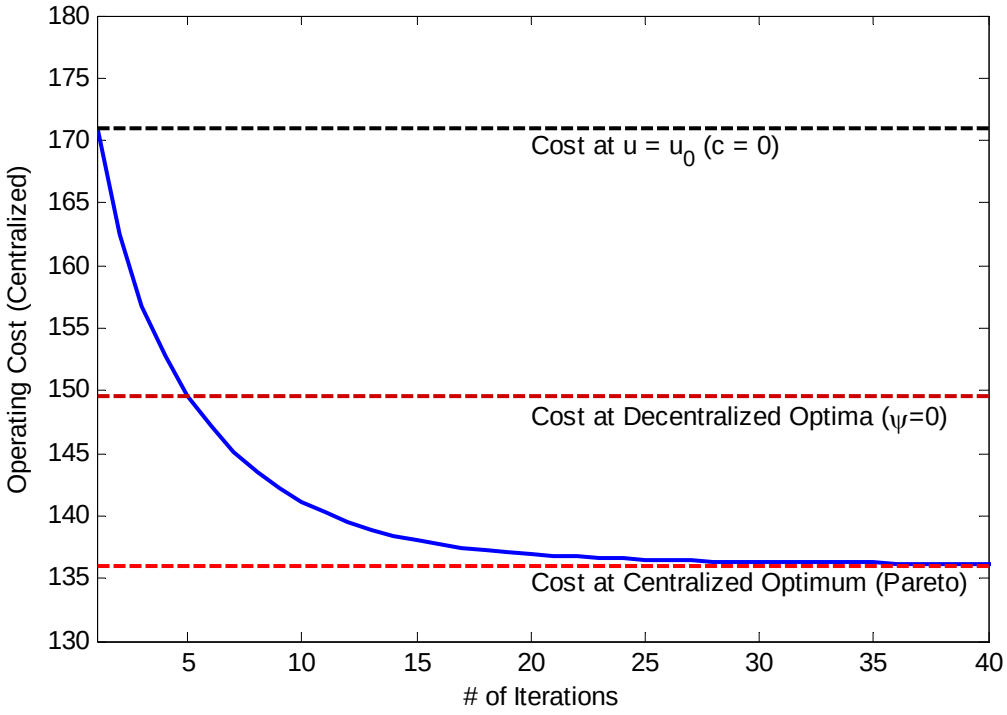


Figure 7.7: Centralized Cost converging to minimum

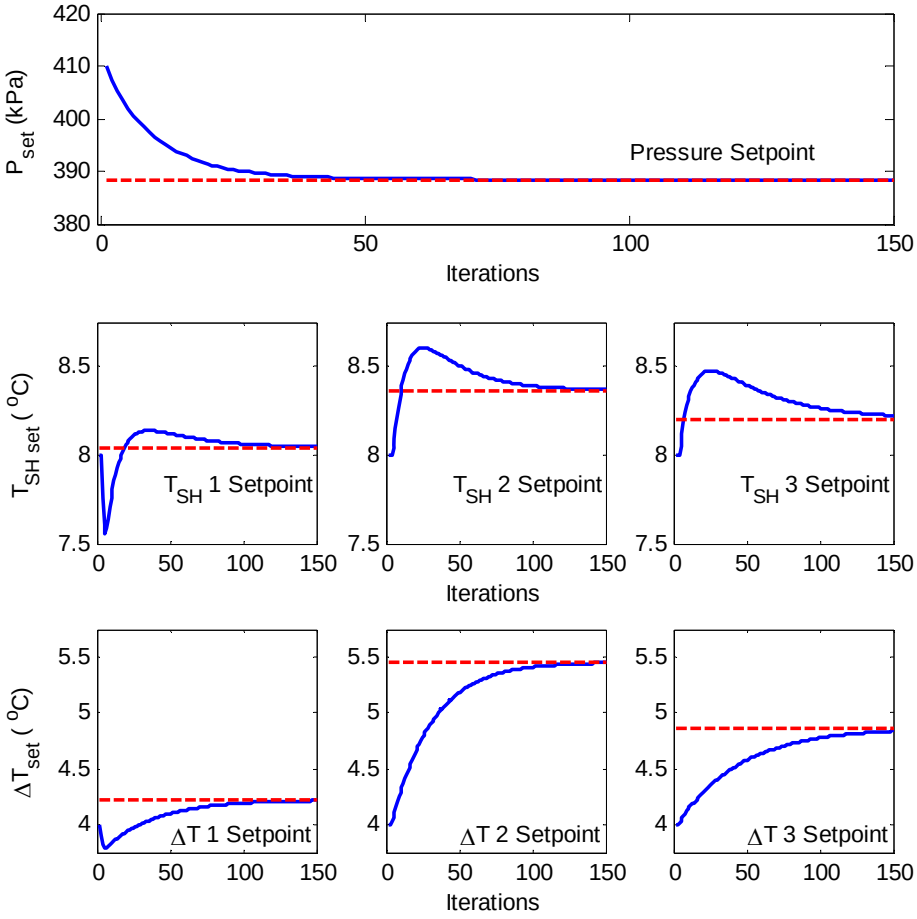


Figure 7.8: Simulation Convergence: Inputs converging to centralized

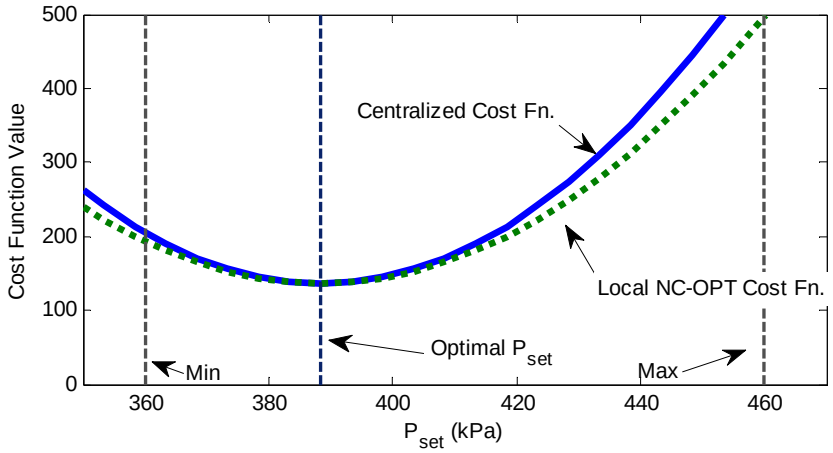


Figure 7.9: Centralized and NC-OPT cost functions around centralized optimum.

Experiments

Comparisons of Iteration Number Effects during a Setpoint Change

The first two experimental tests show the basic function of the NC-OPT architecture. The same suite of optimizers and controllers was used for both tests, with an optimizer time step of 300 seconds. In the first case the optimizers iterated 5 times every time step, and in the second case they iterated 250 times per time step. For both tests, the user-desired Zone 1 temperature setpoint (r_6) was decreased from 22°C to 20°C.

Figure 7.10 compares the setpoints for temperature drop (a) and superheat (b) calculated by the optimizers. In the case with 250 iterations, the centralized optimum is achieved in the first pass, while with 5 iterations per cycle the steady state setpoints are not reached for a long time. The two cases do converge to the same values eventually. Since the dynamics of the zone itself are driven by the superheat and temperature drop values but are considerably slower than that of the VCC system, the higher number of iterations may not be necessary, as is shown in Figure 7.11. This figure shows the dynamic response of Zone 1 temperature for the two cases. Despite the slower setpoint convergence, there is very little difference between the two dynamic responses of the zone temperature. Thus, even if a large number of iterations are required for convergence to an optimum value, slow plant dynamics may allow for a smaller number of iterations per time step.

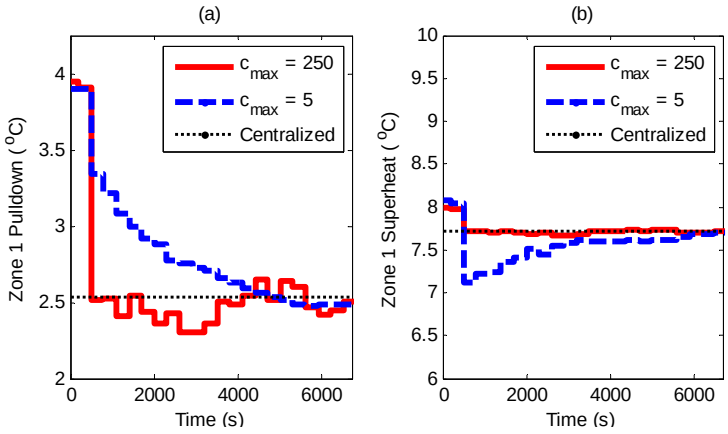


Figure 7.10: Inputs for both tests, compared

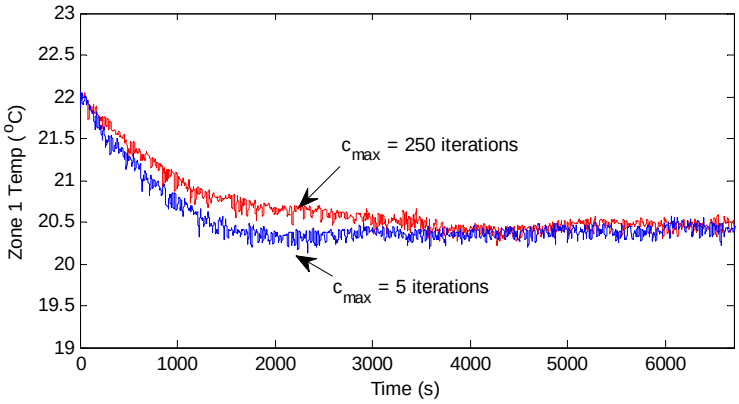


Figure 7.11: System performance, compared

Disturbance Rejection

The next experimental run presented demonstrates the disturbance rejection aspect of the NC-OPT architecture, as well as its ability to tradeoff setpoint tracking criteria. For the test, the heat load disturbance to Zone 3 ($\bar{d}_3$) was increased from approximately 0.55 to 0.75 kW; the output of the Kalman filter is shown in Figure 7.12.

The local controllers respond by lowering the temperature drop setpoint ($\overline{u}_7$) as seen in Figure 7.13(a), which results in increased water flow from the pump. A slight decrease in the superheat setpoint ($\overline{u}_4$) also results, as shown in Figure 7.13(b). These two changes result in more refrigerant flow, and hence more evaporator cooling capacity.

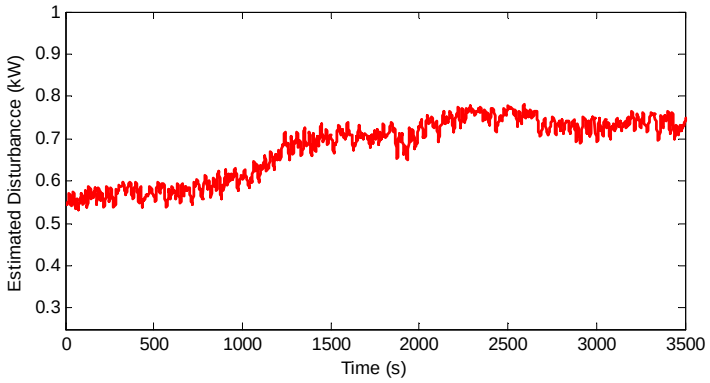


Figure 7.12: Estimated disturbance

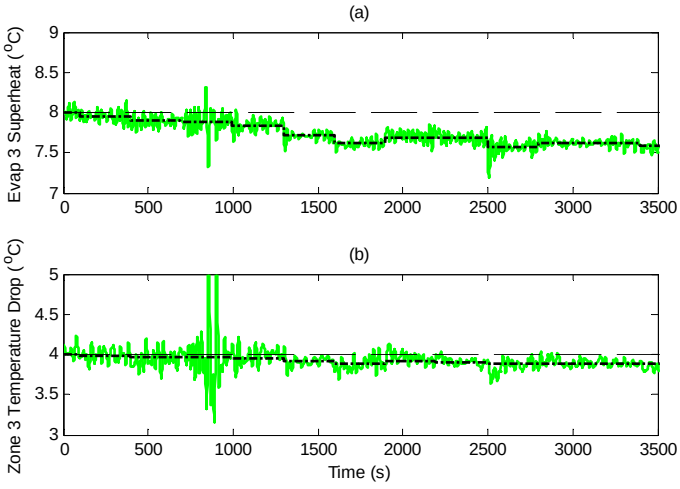


Figure 7.13: Changing input setpoints (u_i) for (a) superheat and (b) temperature drop. User-desired setpoints (r_i) are also shown.

The increased disturbance load means that the system must provide more cooling, and thus consume more power. Since the compressor power consumption is one of the user-defined setpoint (r_2), and since a reduction in power consumption implies the zone temperatures will settle at a higher value, there will be a trade-off between tracking the zone temperature setpoints and the power consumption of the compressor. Figure 7.14 shows how the compressor speed increases to deliver the necessary cooling, as well as the power consumption over the process. Figure 7.15 shows how the zone temperature increases over the process; the accepted steady-state error in the presence of the larger disturbance is larger than the original case. The model-predicted steady-state temperature that would result if the setpoints had remained unchanged under the new disturbance is also shown as a reference. Thus the NC-OPT architecture can be used to balance the comfort demands of the user with the energy and cost reduction desired by building or system operators. This idea is shown further in the next experimental data set.

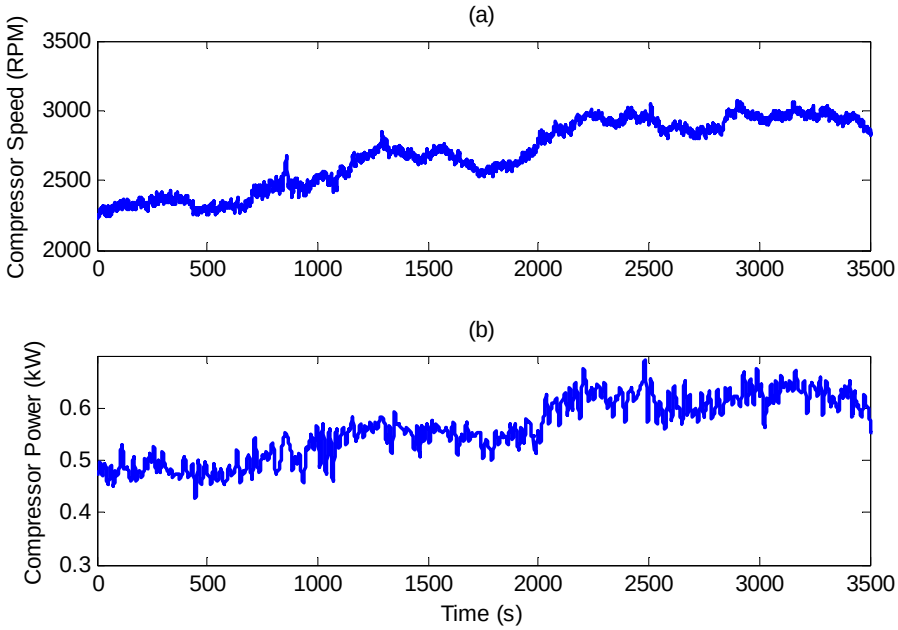


Figure 7.14: (a) Compressor speed and (b) power consumption.

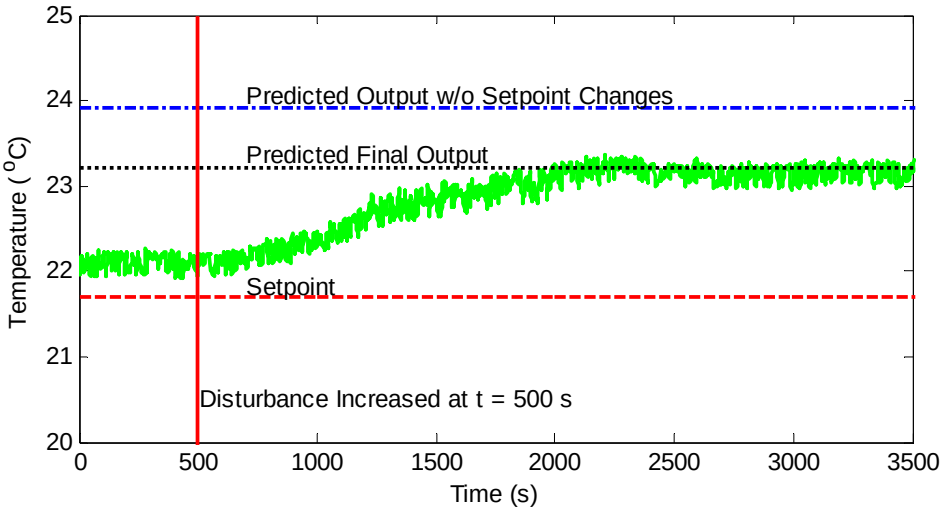


Figure 7.15: Tradeoff between setpoint tracking and power consumption.

Energy Savings

While improved component design has a large effect on system energy consumption, better control of existing control architectures can provide cost savings for a reduced capital investment. This fact is one of the important drivers behind this research.

In this particular implementation of NC-OPT, the compressor power is treated as a setpoint (r_2). The previous data set showed that the system seeks to keep power consumption close to a desired setpoint, but balances this with the zone temperature (comfort) needs of the users. In order to cause the system to actually seek to minimize power consumption, the setpoint r_2 can be set to zero. While a power consumption of zero is obviously not feasible, this setpoint value turns the setpoint tracking problem into an optimization with respect to power consumption.

Two different startup tests were run, wherein the system is turned on and allowed to come to steady state values. For one of the tests, the standard power setpoint of 0.37 kW is used; for the other, a power setpoint of zero. Figure 7.16 shows that the optimizers settle on a higher suction pressure setpoint for the new case, resulting in a slower compressor speed and thus a reduction of approximately 20% in compressor power consumption.

The resulting zone temperature results are given in Figure 7.17. Zones 1 and 2 have large error weights, so the resulting change in zone temperature is small. However, since the error weight on zone three is much smaller compared to the other evaporators, tracking performance for this zone temperature is significantly degraded. This displays

the flexibility of the approach—the weights can be assigned to different ones, permitting a degree of load shedding by the system if necessary to achieve power consumption reduction.

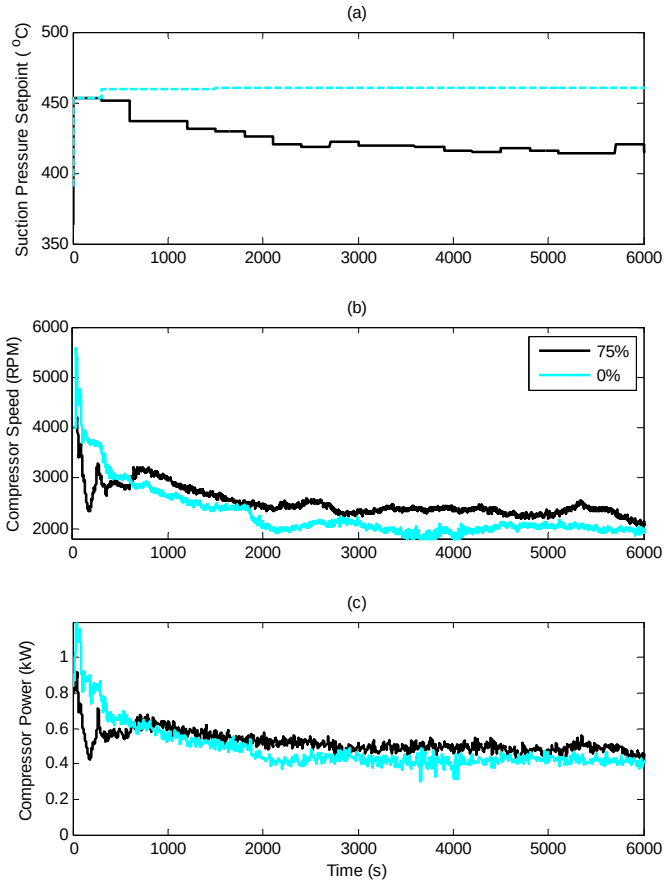


Figure 7.16: (a) Pressure, (b) compressor speed, and (c) power comparisons for two different power setpoints

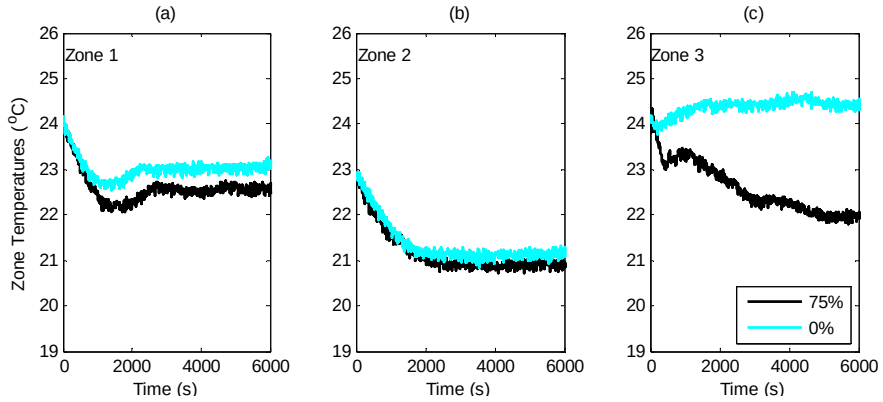


Figure 7.17: Zone temperature tracking comparisons

Alternate Designs

As discussed earlier, the use of integrating controllers at the local level has significant advantages when implementing the NC-OPT architecture; the zero steady-state tracking error exhibited by a PI or PID controller when the actuators do not saturate allows many of the interconnection gains to become zero, simplifying the network topology considerably (compare Figures 7.2 and 7.3). However, the NC-OPT architecture is flexible enough that application using other local level controllers is theoretically possible. For the subsystems chosen for this experimental system non-integrating controllers were not suitable, but a brief discussion is presented for completeness.

Actuator Position-Based NC-OPT

One attempt involved having the NC-OPT optimizers calculate the steady-state value of the actuators themselves; i.e. compressor speed and valve positions. These

actuator positions become the inputs that are fed into the systems. The corresponding block diagrams are shown in Figure 7.18.

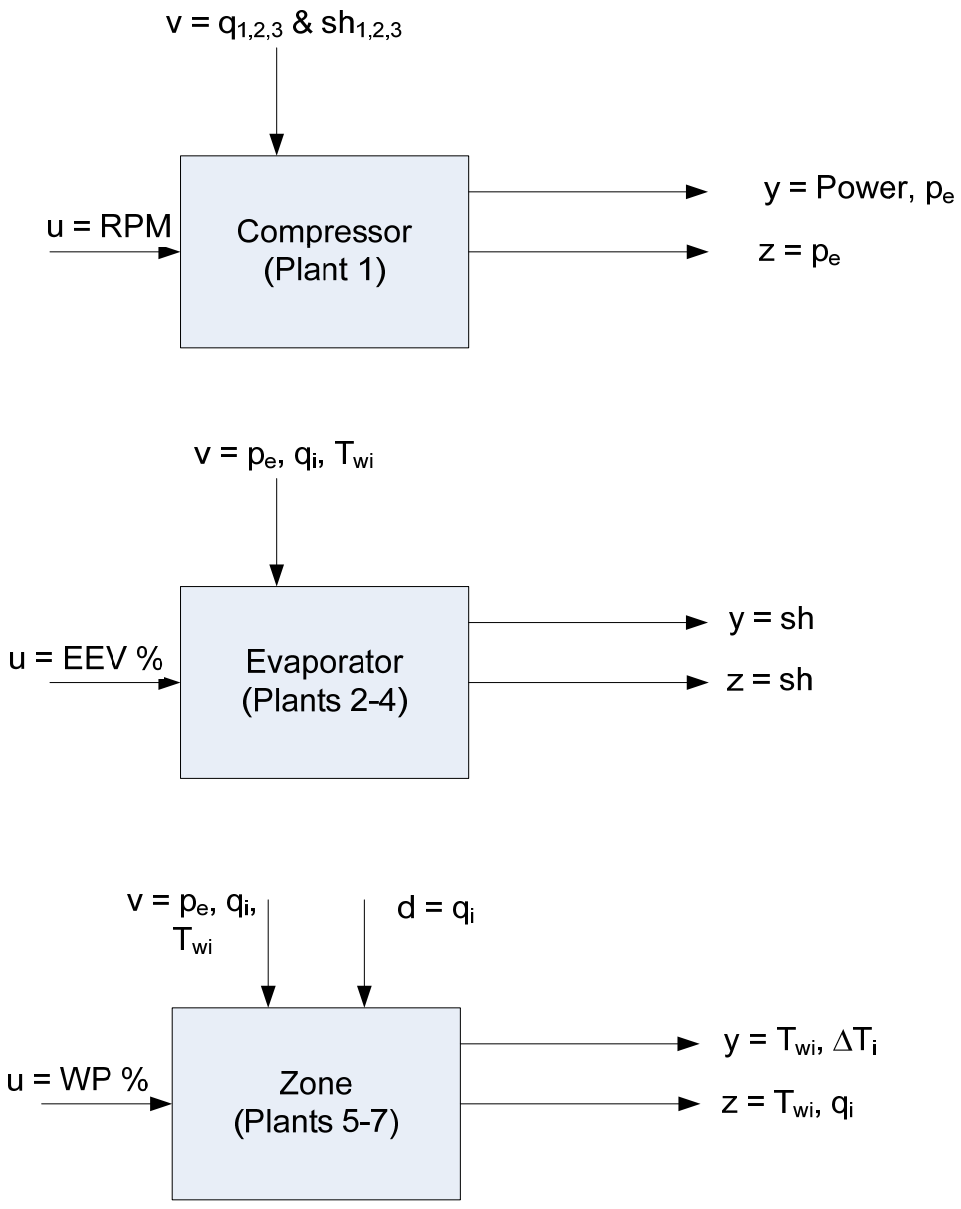


Figure 7.18: Subsystem block diagrams for actuator position-based NC-OPT.

For this implementation, the eigenvalues of the matrix product $\Phi \mathbf{N}_z$ are outside the unit circle on the complex plane, so the information dynamics will not converge, regardless of the value of the convergence term *w*. Figure 7.19 shows the variation in the spectral radius of the state matrix (5.33) as the convergence value changes. The spectral radius is the maximum absolute value of the eigenvalues of the matrix; since it is always greater than 1 for all *w*, there is no value of *w* that will cause the system to converge.

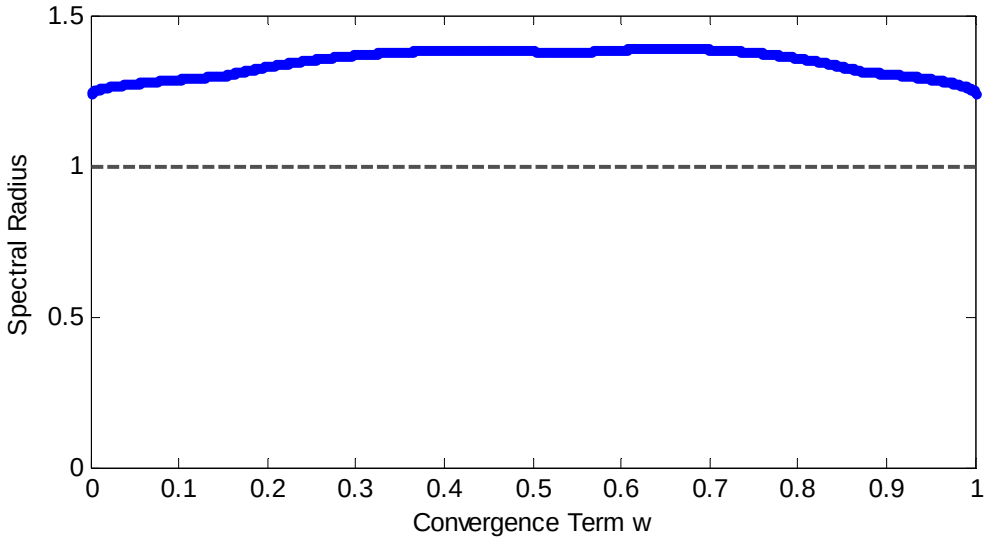


Figure 7.19: Spectral radius of information dynamics state matrix.

Proportional Control-Based NC-OPT

The other implementation considered was a suite of proportional controllers. The structure of the block diagrams would be the same as the PI-controlled case, with different controllers (Figures 7.4, 7.5, and 7.6). However, since proportional controllers tolerate a steady state error, the disturbance rejection from the other plants' outputs would not be complete. Thus the interconnection matrix is the same as the actuator-based NC-OPT and the network topology is the same as well (see Figure 7.3).

An important characteristic of proportional controllers is that the steady state error grows smaller as the controller gain grows larger. This implies that the interconnection gains represented by $\Phi\mathbf{N}_z$ grow smaller as the proportional controller gains grow larger, since the disturbance rejection qualities improve with an increased controller gain. This creates a connection between the dynamics of the subsystem under local control and the behavior of the NC-OPT information dynamics, and can possibly serve as another design parameter when implementing the NC-OPT architecture. For the experimental system used here, however, in order to achieve convergence of the information dynamics the controller gains must be so large that the plant dynamics are rendered closed-loop unstable.

Each of the local controllers were tuned with a proportional controller. At these controller gain values the information dynamics were unstable due to the interconnection gain. To decrease this interconnection gain, the controller gains were increased by a multiplier. Figure 7.20 shows the spectral radius of the matrix product $\Phi\mathbf{N}_z$ as the multiplier ranges from 0 to 10. While using a sufficiently large multiplier will allow the

information dynamics to converge, the controller gains are so large that the controlled systems are unstable. This connection between the two sets of dynamics creates another aspect to the NC-OPT design problem worthy of further study.

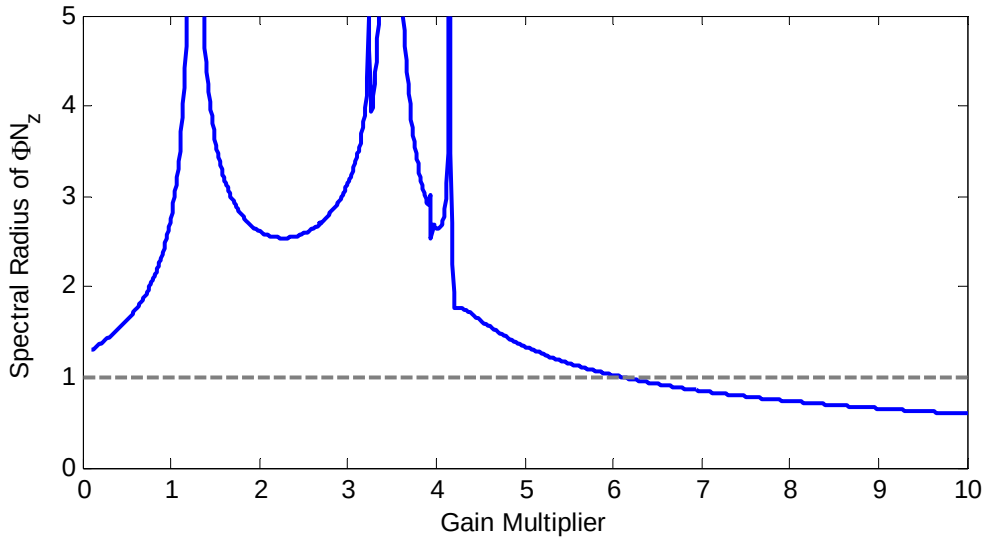


Figure 7.20: Spectral radius of matrix as proportional control gains are increased.

CHAPTER VIII

CONCLUSION

The research presented in this dissertation has the ultimate goal of reducing the energy consumption of air conditioning and refrigeration systems. While improvements in system-level and component design are often the first avenue manufacturers and building managers take to reduce the energy consumption of these systems, improving the control of these components has a strong role to play. This is especially true if better control can be coupled with existing components and systems to improve efficiency while retaining comfort, health, and safety with a minimal increase in capital expenditures.

The work presented herein explored this problem in three different ways. The first was at the component level, with the use of cascaded control of evaporator superheat. This approach allows for tighter control of superheat in the presence of disturbances. As superheat control is improved, a lower setpoint can be employed safely without risking liquid refrigerant being passed to the compressor. Also, the combination of better response and nonlinearity compensation makes this local control architecture a useful tool for controlling complex variable refrigerant flow systems.

The next control technique presented was a hierarchical approach specifically for the control of multiple evaporator vapor compression systems. This technique leverages the capabilities of model predictive control (MPC) to reduce expansion valve wear by allowing superheat to float inside a band. The optimal control aspects of MPC also

allow the control system to and balance energy usage with occupant comfort in air conditioning systems, or with food safety in refrigeration applications. This architecture also allows for an element of modularity in the control system, since each local subsystem is self-contained. This approach still requires a system-level supervisor, however.

In contrast, the final control architecture presented in this dissertation, referred to as neighbor communication optimization (NC-OPT), relies on communication only between "neighbor" subsystems, and subsystems require no knowledge of other systems' dynamics or cost functions. An interconnection matrix guides the communications between optimizers. This is a key departure from other recent distributed MPC approaches that require that all subsystems communicate with each other, contain models of other subsystem dynamics, and use a common system-level cost function to achieve Pareto optimality. An important implication of this is a high level of modularity: an alteration to a particular subsystem does not require any changes to neighboring controllers. Additionally, the construction of the controllers is intuitive and builds upon basic controls knowledge, which is an important practical aspect for inclusion in building systems. Another feature is that this approach allows for the iteration and communication between neighboring subsystems to be decoupled from the implementation of the control action. In essence, the discrete-time dynamics describing the NC-OPT algorithm can evolve separately from (or simultaneously with) the dynamics of the controlled system.

Future Work

Cascaded Control

The cascaded control architecture was applied to an expansion valve in this work. Future studies should explore the effects of this architecture with different components; for example, control of compressor or fan speeds. Additionally, model information can be implemented to develop optimal gains for cascaded controllers; better knowledge of the limitations and usefulness of this approach should be obtainable through optimality studies.

Hierarchical Control of Multi-Evap Systems

There are also opportunities to improve the implementation of the hierarchical control architecture presented in Chapter IV. For example, the efficiency of the system is dependent upon the cooling zone temperature, although this was ignored in the current ICOP function for simplicity. As mentioned earlier, the architecture could also take advantage of more complete models of the zones. Different applications can also be explored, such as a system requiring radically different zone temperatures (e.g., a combination refrigerator/freezer); this will most likely require the use of discharge valves at the exit of higher pressure evaporators.

Neighbor-Communication Optimization

While the calculation of setpoints alone is useful for many applications where the optimizer is coupled with existing dynamic controllers, the extension of the approach to calculate the optimal dynamic path of the system is an important future direction for this work. Additional developments and designs for the architecture for specific systems would also be useful, such as the implementation of coupled constraints. Development of the interconnection matrix for large, complicated systems with hundreds of components remains an open challenge; judicious use of adaptive algorithms might provide a solution to this problem.

From a purely theoretical perspective, developing a scheme to ensure robust stability of the information dynamics is a possibly fruitful line of inquiry. Characterizing the interactions between plant controller dynamics and the information system remains an open problem. One possibility along these lines is simultaneous optimization of the local control gains and the information dynamics. Finally, further development of the algorithm to include stability proofs for hard constraints and nonlinear cost functions and models would also be a welcome development.

Future work in the use of NC-OPT in building systems will include the application of the NC-OPT algorithm to include more elements of building controls, such as humidity and outside air control. Implementation on an actual building on the Texas A&M campus is also in development, including integration with existing building control systems.

REFERENCES

[1] U. S. Dept. of Energy, "Annual Energy Review 2005." Report # DOE/EIA-0384(2005), 2006. Available at http://www.eia.doe.gov/emeu/aer/

[2] U. S. Dept. of Energy Information Administration, "Emissions of Greenhouse Gases in the United States 2008." Report # DOE/EIA-0573(2008) , 2009. Available at http://www.eia.gov/environment/reports.cfm

[3] R. Brown and J. Koomey, "Electricity Use in California: Past Trends and Present Usage Patterns," *Energy Policy,* vol. 31, pp. 849-864, 2003.

[4] J. Rossiter, *Model-Based Predictive Control : A Practical Approach*. CRC Press: Boca Raton, FL, 2003.

[5] E. Camacho and C. Bordons, *Model Predictive Control*. Springer: New York, NY, 2004.

[6] C. Garcia, D. Prett, *et al.*, "Model Predictive Control: Theory and Practice—a Survey," *Automatica,* vol. 25, pp. 335-348, 1989.

[7] S. Qin and T. Badgwell, "A Survey of Model Predictive Control Technology," *Control Engineering Practice,* vol. 11, pp. 733-764, 2003.

[8] D. Clarke, "Generalized Predictive Control--Part I. The Basic Algorithm," *Automatica,* vol. 23, pp. 137-148, 1987.

[9] D. Mayne, J. Rawlings, *et al.*, "Constrained Model Predictive Control: Stability and Optimality," *Automatica,* vol. 36, pp. 789-814, 2000.

[10] M. Kothare, V. Balakrishnan, *et al.*, "Robust Constrained Model Predictive Control Using Linear Matrix Inequalities," *Automatica,* vol. 32, pp. 1361-1379, 1996.

[11] D. Clarke, *Advances in Model-Based Predictive Control*. Oxford University Press: Oxford, UK:, 1994.

[12] J. Rawlings and D. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill: Madison, WI, 2009.

[13] E. Zafiriou and A. Marchal, "Stability of SISO Quadratic Dynamic Matrix Control with Hard Output Constraints," *AIChE Journal,* vol. 37, pp. 1550-1560, 1991.

[14] P. Scokaert and J. Rawlings, "Feasibility Issues in Linear Model Predictive Control," *AIChE Journal,* vol. 45, pp. 1649-1659, 1999.

[15] M. Katebi and M. Johnson, "Predictive Control Design for Large-Scale Systems," *Automatica,* vol. 33, pp. 421-425, 1997.

[16] J. van Antwerp and R. Braatz, "Model Predictive Control of Large Scale Processes," *Journal of Process Control,* vol. 10, pp. 1-8, 2000.

[17] P. Scokaert, D. Mayne*, et al.*, "Suboptimal Model Predictive Control (Feasibility Implies Stability)," *IEEE Transactions on Automatic Control,* vol. 44, pp. 648-654, 1999.

[18] A. Venkat, J. Rawlings*, et al.*, "Stability and Optimality of Distributed Model Predictive Control," from proceedings of the 44th IEEE Conference on Decision and Control, pp. 6680-6685. Seville, Spain, 2005.

[19] Y. Ho, "On Centralized Optimal Control," *IEEE Transactions on Automatic Control,* vol. 50, pp. 537-538, 2005.

[20] N. Sandell, V. Pravin*, et al.*, "Survey of Decentralized Control Methods for Large Scale Systems," *IEEE Transactions on Automatic Control,* vol. 23, pp. 108-128, 1978.

[21] S. Wang and E. Davison, "On the Stabilization of Decentralized Control Systems," *IEEE Transactions on Automatic Control,* vol. 18, pp. 473-478, 1973.

[22] H. Cui and E. Jacobsen, "Performance Limitations in Decentralized Control," *Journal of Process Control,* vol. 12, pp. 485-494, 2002.

[23] D. Siljak, *Decentralized Control of Complex Systems*. Academic Press: Boston, MA, 1991.

[24] L. Bakule, "Decentralized Control: An Overview," *Annual Reviews in Control,* vol. 32, pp. 87-98, 2008.

[25] K. Muske and T. Badgwell, "Disturbance Modeling for Offset-Free Linear Model Predictive Control," *Journal of Process Control,* vol. 12, pp. 617-632, 2002.

[26] L. Magni and R. Scattolini, "Stabilizing Decentralized Model Predictive Control of Nonlinear Systems," *Automatica,* vol. 42, pp. 1231-1236, 2006.

[27] J. Rawlings and B. Stewart, "Coordinating Optimization-Based Controllers: New Oppotunities and Challenges," *Journal of Process Control,* vol. 18, pp. 839-845, 2008.

[28] D. Jia and B. H. Krogh, "Distributed Model Predictive Control," from proceedings of the 2001 American Control Conference, pp. 2767-2772. Arlington, VA, 2001.

[29] D. Jia and B. H. Krogh, "Min-Max Feedback Model Predictive Control for Distributed Control with Communication," from proceedings of the 2002 American Control Conference, pp.4507-4512. Anchorage, AK, 2002.

[30] E. Camponogara, D. Jia*, et al,*. "Distributed Model Predictive Control," *Control Systems Magazine*, February 2002, pp. 44-52.

[31] M. Mercangoz and F. Doyle, "Distributed Model Predictive Control of an Experimental Four-Tank System," *Journal of Process Control,* vol. 17, pp. 297-308, 2007.

[32] K. Johansson and J. Nunes, "A Multivariable Laboratory Process with an Adjustable Zero," from proceedings of the 1998 American Control Conference, pp. 2045-2049. Philadelphia, PA, 1998.

[33] E. Camponogara and L. de Oliveira, "Distributed Optimization for Model Predictive Control of Linear-Dynamic Networks," *IEEE Transactions on Systems, Man, and Cybernetics--Part A: Systems and Humans,* vol. 39, pp. 1331-1338, 2009.

[34] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, 2nd ed. SIAM: Philadelphia, PA, 1999.

[35] A. N. Venkat, J. Rawlings*, et al.*, "Distributed Model Predictive Control of Large-Scale Systems," in *Assessment and Future Directions of Nonlinear Model Predictive Control*, pp. 591-605. R. Findeisen, F. Allgöwer*, et al.*, Editors. Springer: Berlin, Germany, 2007.

[36] B. Stewart, A. Venkat, *et al.*, "Cooperative Distributed Model Predictive Control," *Systems & Control Letters,* vol. 59, pp. 460-469, 2010.

[37] B. Stewart, S. Wright*, et al.*, "Cooperative Distributed Model Predictive Control for Nonlinear Systems," *Journal of Process Control,* vol. 21, pp. 698-704, 2011.

[38] J. Maestre, D. de la Peña, *et al.*, "Distributed Model Predictive Control Based on a Cooperative Game," *Optimal Control Applications and Methods,* vol. 32, pp. 153-176, 2011.

[39] J. Maestre, D. de la Peña, *et al.*, "Distributed Model Predictive Control Based on Agent Negotiation," *Journal of Process Control,* vol. 21, pp. 685-697, 2011.

[40] A. Ferramosca, D. Limon, *et al.*, "Cooperative Distributed MPC for Tracking," *Automatica,* vol. 49, pp. 906-914, 2013.

[41] B. Stewart, J. Rawlings, *et al.*, "Hierarchical Cooperative Distributed Model Predictive Control," from proceedings of the 2010 American Control Conference, pp. 3963-3968. Baltimore, MD, 2010.

[42] T. Meadowcroft, G. Stephanopoulos, *et al.*, "The Modular Multivariable Controller: 1: Steady-State Properties," *AIChE Journal,* vol. 38, pp. 1254-1278, 1992.

[43] W. Dunbar and R. Murray, "Distributed Receding Horizon Control for Multi-Vehicle Formation Stabilization," *Automatica,* vol. 42, pp. 549-558, 2006.

[44] W. Dunbar, "Distributed Receding Horizon Control of Dynamically Coupled Nonlinear Systems," *IEEE Transactions on Automatic Control,* vol. 52, pp. 1249-1263, 2007.

[45] J. Liu, D. de la Peña, *et al.*, "Distributed Model Predictive Control of Nonlinear Process Systems," *AIChE Journal,* vol. 55, pp. 1171-1184, 2009.

[46] T. Keviczky, F. Borrelli, *et al.*, "Decentralized Receding Horizon Control for Large Scale Dynamically Decoupled Systems," *Automatica,* vol. 42, pp. 2105-2115, 2006.

[47] G. Zhu and M. Henson, "Model Predictive Control of Interconnected Linear and Nonlinear Processes," *Industrial & Chemical Engineering Research,* vol. 41, pp. 801-816, 2002.

[48] R. Cheng, J. Forbes, *et al.*, "Dantzig–Wolfe Decomposition and Plant-Wide MPC Coordination," *Computers & Chemical Engineering,* vol. 32, pp. 1507-1522, 2008.

[49] C. Ocampo-Martínez, S. Bovo, *et al.*, "Partitioning Approach Oriented to the Decentralised Predictive Control of Large-Scale Systems," *Journal of Process Control,* vol. 21, pp. 775-786, 2011.

[50] A. Alessio, D. Barcelli, *et al.*, "Decentralized Model Predictive Control of Dynamically Coupled Linear Systems," *Journal of Process Control,* vol. 21, pp. 705-714, 2011.

[51] I. Necoara, V. Nedelcu, *et al.*, "Parallel and Distributed Optimization Methods for Estimation and Control in Networks," *Journal of Process Control,* vol. 21, pp. 756-766, 2011.

[52] P. Christofides, R. Scattolini, *et al.*, "Distributed Model Predictive Control: A Tutorial Review and Future Research Directions," *Computers & Chemical Engineering,* vol. 51, pp. 21-41, 2013.

[53] Y. Zhang and S. Li, "Networked Model Predictive Control Based on Neighbourhood Optimization for Serially Connected Large-Scale Processes," *Journal of Process Control,* vol. 17, pp. 37-50, 2007.

[54] M. Farina and R. Scattolini, "Distributed Predictive Control: A Non-Cooperative Algorithm with Neighbor-to-Neighbor Communication for Linear Systems," *Automatica,* vol. 48, pp. 1088-1096, 2012.

[55] H. Scheu and W. Marquardt, "Sensitivity-Based Coordination in Distributed Model Predictive Control," *Journal of Process Control,* vol. 21, pp. 715-728, 2011.

[56] R. Scattolini, "Architectures for Distributed and Hierarchical Model Predictive Control--A Review," *Journal of Process Control,* vol. 19, pp. 723-731, 2009.

[57] M. He, W. Cai, *et al.*, "Multiple Fuzzy Model-Based Temperature Predictive Control for HVAC Systems," *Information Sciences,* vol. 169, pp. 155-174, 2005.

[58] M. Xu, S. Li, *et al.*, "Effects of a GPC-PID Control Strategy with Hierarchical Structure for a Cooling Coil Unit," *Energy Conversion and Management,* vol. 46, pp. 132-145, 2006.

[59] M. Xu, S. Li, *et al.*, "Practical Receding Horizon Optimization Control of the Air Handling Unit in HVAC Systems," *Industrial & Chemical Engineering Research,* vol. 44, pp. 2848-2855, 2005.

[60] J. MacArthur and M. Woessner, "Receding Horizon Control: A Model-Based Policy for HVAC Applications," *ASHRAE Transactions,* vol. 99, pp. 139-148, 1993.

[61] A. Dexter and P. Haves, "A Robust Self-Tuning Predictive Controller for HVAC Applications," *ASHRAE Transactions,* vol. 95, pp. 431-437, 1989.

[62] J. Sousa, J. Babuska, *et al.*, "Fuzzy Predictive Control Applied to an Air Conditioning System," *Control Engineering Practice,* vol. 5, pp. 1395-1406, 1997.

[63] X. Xi, A. Poo, *et al.*, "Support Vector Regression Model Predictive Control on a HVAC Plant," *Control Engineering Practice,* vol. 15, pp. 897-908, 2007.

[64] S. Yuan and R. Perez, "Model Predictive Control of Supply Air Temperature and Outside Air Intake Rate of a VAV Air-Handling Unit," *ASHRAE Transactions,* vol. 112, pp. 145-161, 2006.

[65] G. Huang and S. Wang, "Two-Loop Robust Model Predictive Control for the Temperature Control of Air-Handling Units," *HVAC&R Research,* vol. 14, pp. 565-580, July 2008.

[66] G. Huang, S. Wang, *et al.*, "Robust Model Predictive Control of VAV Air-Handling Units Concerning Uncertainties and Constraints," *HVAC&R Research,* vol. 16, pp. 15-33, January 2010.

[67] G. Huang, S. Wang, *et al.*, "A Robust Model Predictive Control Strategy for Improving the Control Performance of Air Conditioning Systems," *Energy Conversion and Management,* vol. 50, pp. 2650-2658, 2009.

[68] X. Xu, S. Wang, *et al.*, "Robust MPC for Temperature Control of Air Conditioning Systems Concerning on Constraints and Multitype Uncertainties," *Building Services Engineering Research & Technology,* vol. 31, pp. 39-55, 2010.

[69] B. Paris, J. Eynard, *et al.*, "Heating Control Schemes for Energy Management in Buildings," *Energy and Buildings,* vol. 42, pp. 1908-1917, 2010.

[70] S. Privara, J. Siroky, *et al.*, "Model Predictive Control of a Building Heating System: The First Experience," *Energy and Buildings,* vol. 43, pp. 564-572, 2011.

[71] P. May-Ostendorp, G. Hinze, *et al.*, "Model Predictive Control of Mixed-Mode Buildings with Rule Extraction," *Building and Environment,* vol. 46, pp. 428-437, 2011.

[72] R. Freire, G. Oliveira, *et al.*, "Predictive Controllers for Thermal Comfort Optimization and Energy Savings," *Energy and Buildings,* vol. 40, pp. 1353-1365, 2008.

[73] D. Leducq, J. Guilpart, *et al.*, "Non-linear Predictive Control of a Vapour Compression Cycle," *International Journal of Refrigeration,* vol. 29, pp. 761-772, 2006.

[74] C. Changenet, J. Charvet, *et al.*, "Study on Predictive Functional Control of an Expansion Valve for Controlling the Evaporator Superheat," *Proceedings of the Institution of Mechanical Engineers--Part I--Journal of Systems & Control Engineering,* vol. 222, pp. 571-582, 2008.

[75] H. Fallahsohi, C. Changenet, *et al.*, "Predictive Functional Control of an Expansion Valve for Minimizing the Superheat of an Evaporator," *International Journal of Refrigeration,* vol. 33, pp. 409-418, 2010.

[76] Y. Ma, G. Anderson, *et al.*, "A Distributed Predictive Control Approach to Building Regulation," from proceedings of the 2011 American Controls Conference, pp. 2089-2094. San Francisco, CA, 2011.

[77] P. Morosan, R. Bourdais, *et al.*, "Building Temperature Regulation Using a Distributed Model Predictive Control," *Energy and Buildings,* vol. 42, pp. 1445-1452, 2010.

[78] P. Morosan, R. Bourdais, *et al.*, "A Distributed MPC Strategy Based on Benders' Decomposition Applied to Multi-Source Multi-Zone Temperature Regulation," *Journal of Process Control,* vol. 21, pp. 729-737, 2011.

[79] R. Bourdais and H. Guéguen, "Distributed Predictive Control for Complex Hybrid System. The Refrigeration System Example," from proceedings of the 12th IFAC Symposium on Large Scale Systems, pp. 388-394. Villeneuve d'Ascq, France, 2010.

[80] W. Goetzler, K. Roth, *et al.,* "Variable Flow and Volume Refrigerant System," *ASHRAE Journal Magazine,* January 2004, pp.164-165.

[81] M. He, S. Liu, *et al.*, "Modeling of Vapor Compression Cycles for Multivariable Feedback Control of HVAC Systems," *Journal of Dynamic Systems, Measurement, and Control,* vol. 119, pp. 183-191, 1997.

[82] T. Aynur, Y. Hwang, *et al.*, "Simulation Comparison of VAV and VRF Air Conditioning Systems in an Existing Building for the Cooling Season," *Energy and Buildings,* vol. 41, pp. 1143-1150, 2009.

[83] M. Elliott and B. Rasmussen, "Model-Based Predictive Control of a Multi Evaporator Vapor Compression Cooling Cycle," from proceedings of the 2008 American Control Conference, pp. 1463-1468. Seattle, WA, 2008.

[84] M. Elliott and B. Rasmussen, "A Model-Based Predictive Supervisory Controller for Multi Evaporator HVAC systems," from proceedings of the 2009 American Control Conference, pp. 3669-3674. St. Louis, MO, 2009.

[85] J. Choi and Y. Kim, "Capacity Modulation of an Inverter-Driven Multi-Air Conditioner Using Electronic Expansion Valves," *Energy,* vol. 28, pp. 141-155, 2003.

[86] Y. Park, Y. Kim*, et al.*, "Performance Analysis on a Multi-type Inverter Air Conditioner," *Energy Conversion and Management,* vol. 42, pp. 1607-1621, 2001.

[87] X. He and H. Asada, "A New Feedback Linearization Approach to Advanced Control of Multi-unit HVAC Systems," from proceedings of the 2003 American Control Conference, pp. 2311-2316. Denver, CO, 2003.

[88] G. Lee, M. Kim*, et al.*, "An Experimental Study of the Capacity Control of Multi-Type Heat Pump System Using Predictive Control Logic," from proceedings of the 7th International Energy Agency Heat Pump Conference, pp. 158-165. Beijing, China, 2002.

[89] R. Shah, A. Alleyne*, et al.*, "Dynamic Modeling and Control of Multi-Evaporator Air Conditioning Systems," *ASHRAE Transactions,* vol. 110, pp. 109-119, 2004.

[90] C. B. Chiou, C. H. Chiou*, et al.*, "The Application of Fuzzy Control on Energy Saving for Multi-Unit Room Air Conditioners," *Applied Thermal Engineering,* vol. 29, pp. 310-316, 2009.

[91] T. Aynur, "Variable Refrigerant Flow Systems: A Review," *Energy and Buildings,* vol. 42, pp. 1106-1112, 2010.

[92] S. Wang and Z. Ma, "Supervisory and Optimal Control of Building HVAC Systems: A Review," *HVAC&R Research,* vol. 14, pp. 3-32, 2008.

[93] J. Bourdouxhe, M. Grodent*, et al.*, *Reference Guide for Dynamic Models of HVAC Equipment*. ASHRAE: Atlanta, GA, 1998.

[94] S. Bendapudi and J. E. Braun, "A Review of Literature on Dynamic Models for Vapor Compression Equipment." ASHRAE Report #4036-5, 2002. Available at http://www.nist.gov/tc75/content.html

[95] B. Rasmussen, "Dynamic Modeling for Vapor Compression Systems—Part I: Literature Review," *HVAC&R Research,* vol. 18, pp. 934-955, 2011.

[96] B. Rasmussen and B. Shenoy, "Dynamic Modeling for Vapor Compression Systems—Part II: Simulation Tutorial," *HVAC&R Research,* vol. 18, pp. 956-973, 2011.

[97] B. Rasmussen and A. Alleyne, "Control-Oriented Modeling of Transcritical Vapor Compression Systems," *Journal of Dynamic Systems, Measurement, and Control,* vol. 126, pp. 54-64, 2004.

[98] E. Grald and J. MacArthur, "A Moving-Boundary Formulation for Modeling Time-Dependent Two-Phase Flows," *International Journal of Heat and Fluid Flow,* vol. 13, pp. 266-272, 1992.

[99] G. Wedekind, B. Bhatt*, et al.*, "A System Mean Void Fraction Model for Predicting Various Transient Phenomena Associated with Two-Phase Evaporating and Condensing Flows," *International Journal of Multiphase Flow,* vol. 4, pp. 97-114, 1978.

[100] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Press: Cambridge, UK, 2004.

[101] M. Elliott, Z. Walton*, et al.*, "Superheat Control: A Hybrid Approach," *HVAC&R Research,* vol. 15, pp. 1021-1044, 2009.

[102] D. Juričić, S. Strmčnik*, et al.,* "Compensator for Static Nonlinearities: Design and Application," *Electronics Letters* vol. 22*,* pp. 532-534, 1986.

[103] A. Singhal and T. Salsbury, "Characterization and Cancellation of Static Nonlinearity in HVAC Systems," *ASHRAE Transactions*, vol. 113, p. 391, 2007.

[104] K. Ogata, *Modern Control Engineering*, 4th ed. Prentice Hall: Upper Saddle River, NJ, 2002.

[105] M. Elliott and B. Rasmussen, "Evaporator Superheat Regulation Via Emulation of Semi-active Flow Control," from proceedings of the 2009 Dynamic Systems and Controls Conference, pp. 465-471. Hollywood, CA, 2009.

[106] M. Elliott, C. Estrada*, et al.*, "Cascaded Superheat Control with a Multiple Evaporator Refrigeration System," from proceedings of the 2011 American Control Conference , pp. 2065-2070. San Francisco, CA, 2011.

[107] M. S. Elliott, B. Shenoy*, et al.*, "A Control Architecture Solution to Superheat Nonlinearity," from proceedings of the 2010 American Control Conference, pp. 5898-5903. Baltimore, MD, 2010.

[108] M. Elliott and B. Rasmussen, "On Reducing Evaporator Superheat Nonlinearity with Control Architecture," *International Journal of Refrigeration,* vol. 33, pp. 607-614, 2010.

[109] M. S. Elliott, "Decentralized model predictive control of a multiple evaporator HVAC system," Master of Science Thesis, Texas A&M University. College Station, TX, 2008.

[110] M. Moran and H. Shapiro, *Fundamentals of Engineering Thermodynamics*, 3rd ed. John Wiley & Sons: New York, NY, 1996.

[111] A. Gupta, "Reduced Order Modeling of Heat Exchangers using High Order Finite Control Volume Models." Master of Science Record of Study, Texas A&M University. College Station, TX, 2007.

[112] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Prentice Hall: Upper Saddle River, NJ, 1999.

[113] H. Li and J. Braun, "Decoupling of Features and Virtual Sensors for Diagnosis of Faults in Vapor Compression Air Conditioners," *International Journal of Refrigeration,* vol. 30, pp. 546-564, 2007.

[114] J. Maciejowski, *Predictive Control with Constraints*. Prentice Hall: London, UK, 2000.

[115] A. Bemporad, M. Moriari, *et al.*, *MATLAB Model Predictive Control Toolbox Manual*. The Mathworks: Natick, MA, 2006.

[116] U. S. Dept. of. Energy. *EnergyPlus Energy Simulation Software*. Available at: http://apps1.eere.energy.gov/buildings/energyplus/

[117] Trimble Buildings. *SketchUp Software*. Available: http://www.sketchup.com/

[118] U. S. Dept. of. Energy. *OpenStudio Software*. Available at: http://apps1.eere.energy.gov/buildings/energyplus/openstudio.cfm

[119] Lawrence Berkeley National Lab.. *Building Controls Virtual Test Bed Software*. Available at: http://simulationresearch.lbl.gov/bcvtb

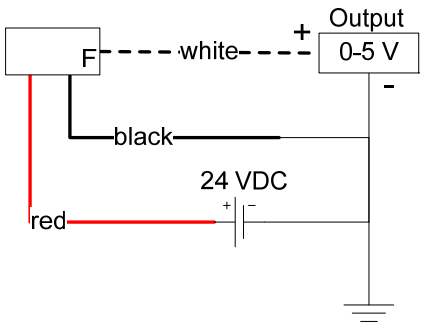
[120] Embedded Systems Lab., U. of Pennsylvania. *MLE+ Software*. Available at: http://mlab.seas.upenn.edu/mlep/

[121] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific: Belmost, MA, 1997.

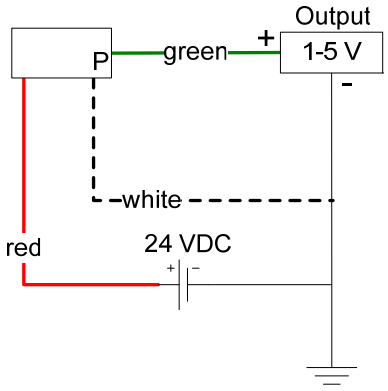
[122] M. Elliott, E. Jones, *et al.*, "A Study of Evaporator Superheat Dynamics, Efficiency Degradation, and Control," *International Journal of Refrigeration,* in review.

APPENDIX

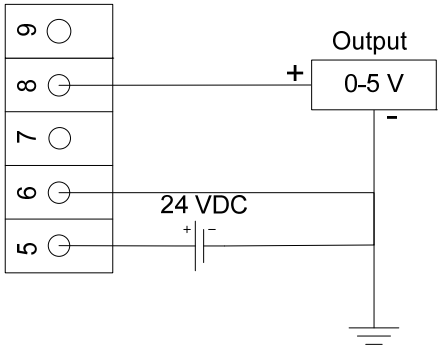
McMillan 102-6 Sensor: Refrigerant Flow



Cole Parmer 03756 Sensor: Refrigerant Pressure



CRM CR5210 Sensor: Compressor Current



Notes:

1. Wiring is the same for high and low pressure transducers.
2. Colors, when given, are the wire colors on the attached sensor leads.
3. Outputs are routed to a standard BNC connection.

Fig. A.1 Sensor wiring schematics

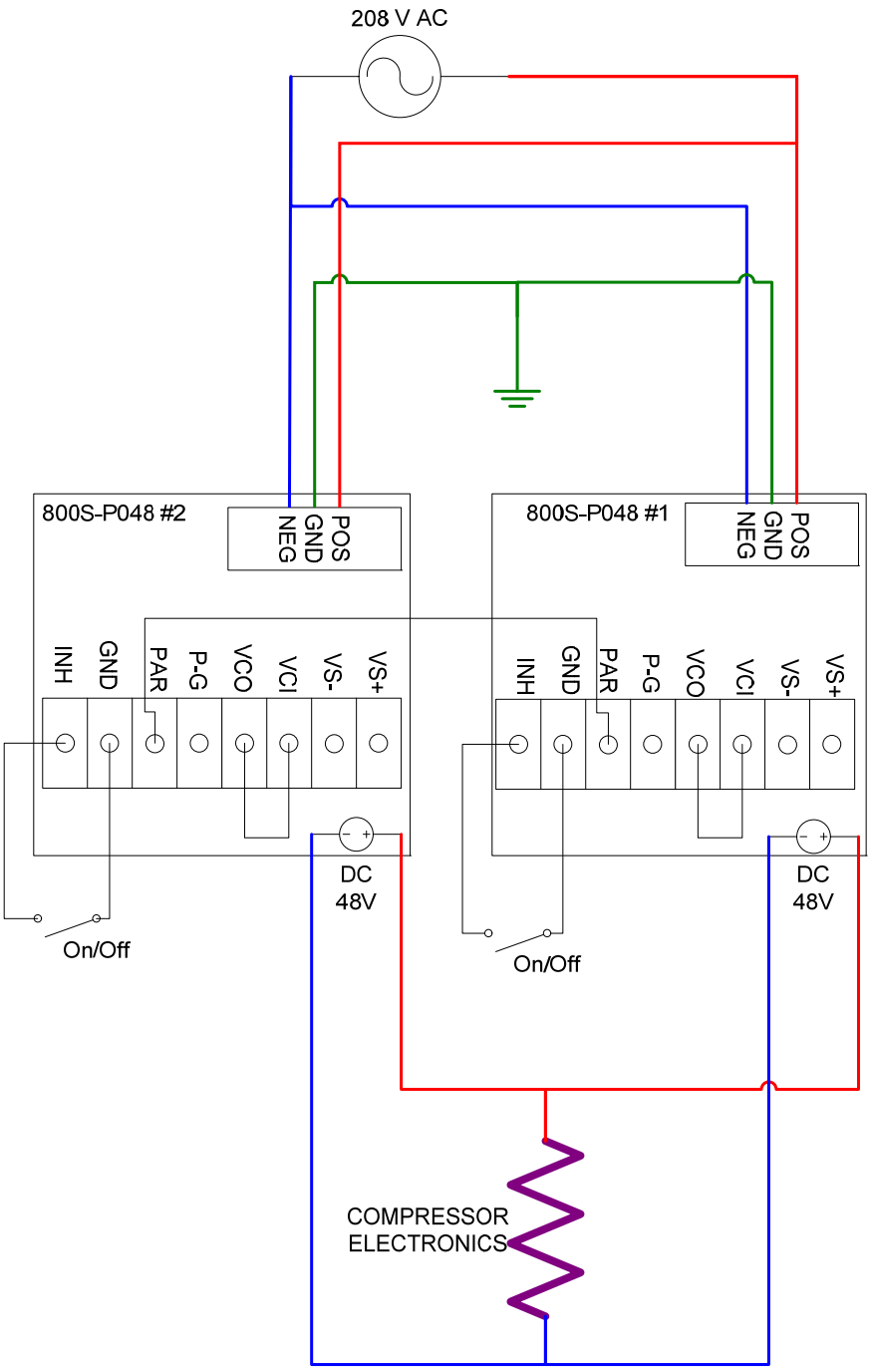
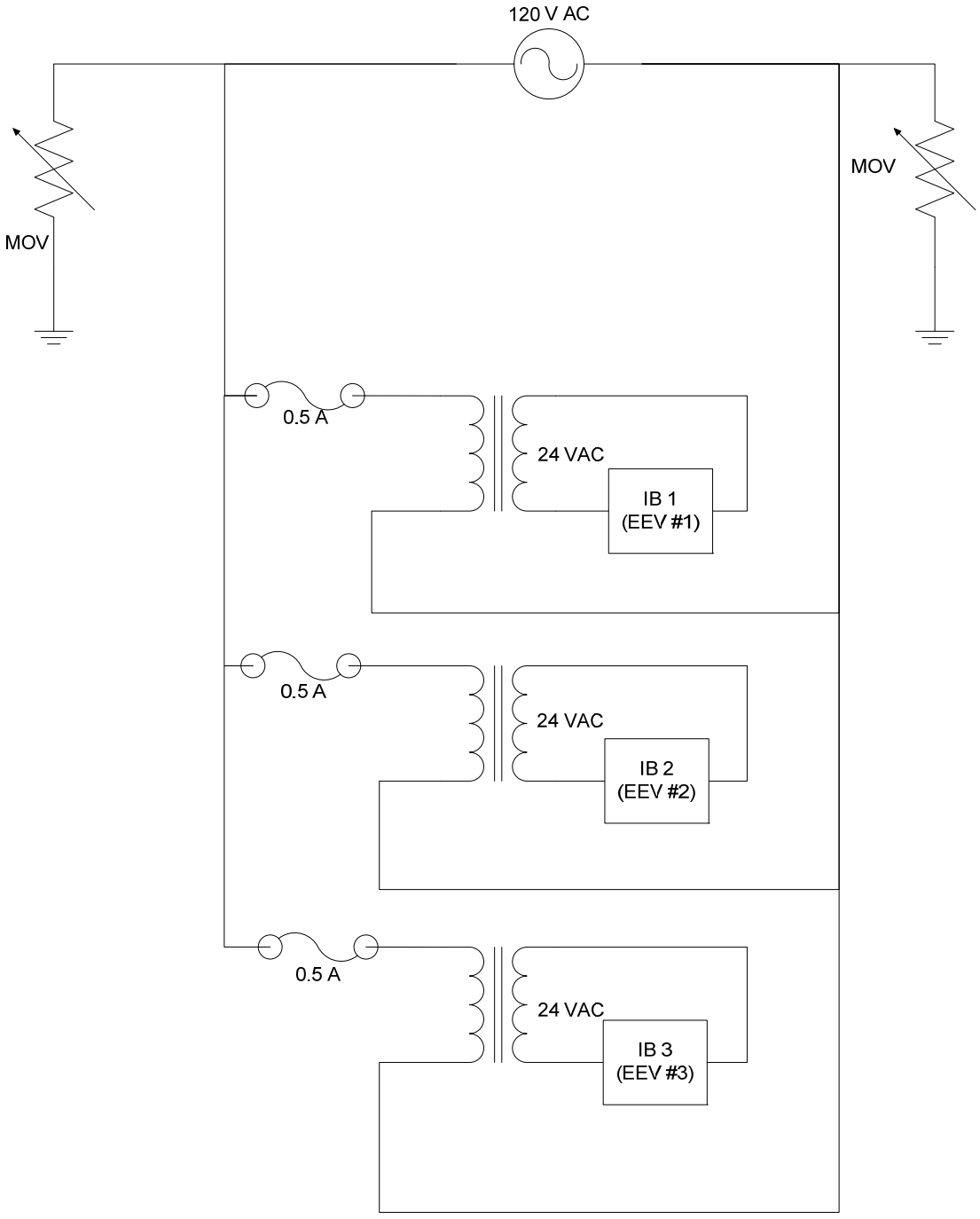


Fig. A.2 Compressor power wiring



Notes:
1. MOV = Metal Oxide Varistor; supplied by Sporlan (IB Mfr.)

Fig. A.3 120VAC wiring

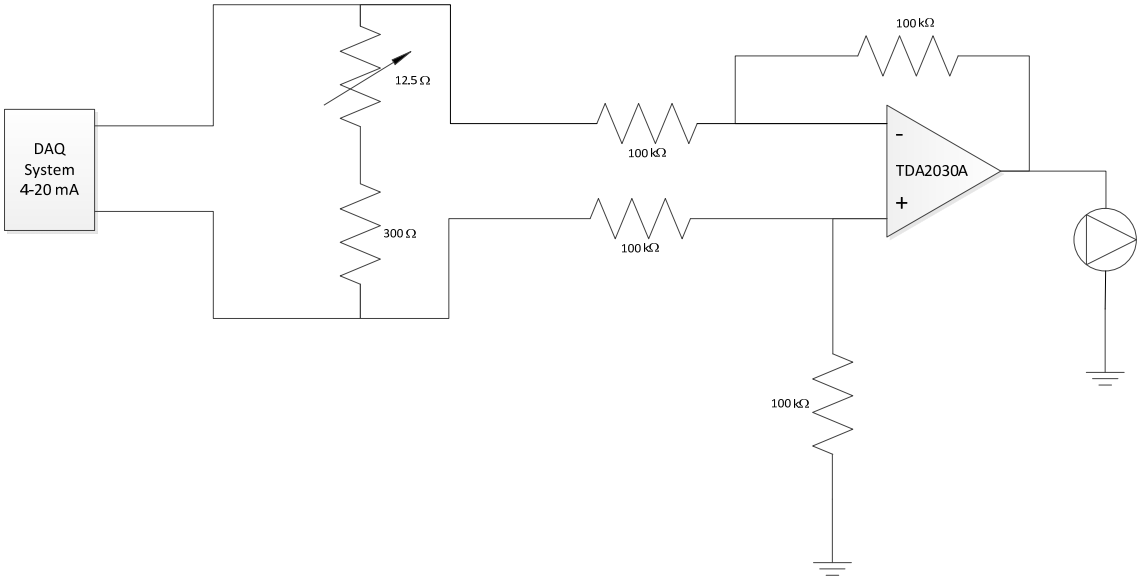


Figure A.4: Water Pump Amplifier Circuit

Table A.1: Experimental System Transducers

Description	Qty	Mfr.	Part Number	Operating Range	Output	Listed Accuracy, +/-
Thermocouples	16	Omega	GTMQSS-062U-6	-270-400 °C	TC	0.5 °C
Suction Pressure	1	Cole-Parmer	07356-03	0-100 psig	1-5 V	1.0% FS
Condenser Pressure	1	Cole-Parmer	07356-04	0-300 psig	1-5 V	1.0% FS
Refrigerant Flow	3	McMillan	102-6-E-Q-B4-NIST	100-1000 mL/min	0-5 V	3.0% FS
Compressor Current	1	CR Magnetics	CR5210	0-50 amps DC	0-5 V	1.0% FS

Table A.2: Experimental System Power Components

Description	Qty	Mfr.	Part Number	Components Powered	Rating
Transformer, 24 VAC	3	White Rodgers	T40-24M1	Provides power for EEVs	40VA
Power Supply, 48 VDC	2	Cotek	800S-P048	Compressor; requires 208VAC suuply	800W
Power Supply, 24 VDC	1	Traco	TML 15124C	Sensors	15W
Power Supply, 12 VDC	1	TDK-Lambda	LS150-15	Evaporator Water Pumps	150W
Power Supply, 12 VDC	1	TDK-Lambda	LS100-12	Condenser Water Pumps, Cooling Fans, Disturbance Pumps, Water Chiller Pump	100W
Power Supply, 5 VDC	1	Traco	TML 15105C	Signal Conditioning	15W

Table A.3: Experimental System Signal Conditioning Modules

Signal	Mfr.	Part Number	Input	Output
EEV 1	Dataforth	SCM5B39-01	0-5V	4-20 ma
EEV 2	Dataforth	SCM5B39-01	0-5V	4-20 ma
EEV 3	Dataforth	SCM5B39-01	0-5V	4-20 ma
WP 1	Dataforth	SCM5B39-01	0-5V	4-20 ma
WP 2	Analog Devices	5B39-01	0-5V	4-20 ma
WP 3	Analog Devices	5B39-01	0-5V	4-20 ma
Compressor	Omega	OM5-IV-10B-C	-10 to +10V	0-5V
Unused	Omega	OM5-IV-10B-C	-10 to +10V	0-5V

Table A.4: Actuators

Actuator	Mfr.	Model No.	Range	Control Signal
Compressor	Masterflux	SIERRA03-0982Y3	1800-6500 RPM	0-5 VDC; Manual switch
EEV	Sporlan	SEI 0.5-10'-S	0-100%	4-20 mA
Evaporator Water Pump	Swiftech	MCP350	0-100%	4-20 mA
Condenser Water Pump	Swiftech	MCP655	ON/OFF	Manual switch
Chiller Water Pump	Swiftech	MCP655	ON/OFF	0/5V Digital
Disturbance Water Pump	Swiftech	MCP350	ON/OFF	0/5V Digital

Table A.5: Miscellaneous components

Description	Qty	Manufacturer	Part Number
PRIMARY LOOP			
Evaporator	3	Packless Ind.	COAX-2051-H-07-51
Condenser	1	Packless Ind.	COAX-2151-H-08-110
Condenser Chiller	1	Packless Ind.	COAX-2301-J-10-144
Manual Shutoff Valve, 1/4"	9	Mueller	A14833
Manual Shutoff Valve, 3/8"	4	Mueller	A14835
Liquid Receiver	1	Henry Technologies	S-8060
Sight Glass	1	Emerson	AMI 1FM2
Filter Drier	1	Alco	EK-032
SECONDARY LOOP			
Condenser Water Tanks	2	Tamco	6314
Evaporator Water Tanks	3	Tamco	6305
DATA ACQUISITION BOARDS			
Thermocouple Board	1	Measurement Computing	PCI-DAS-TC
Analog Output Board	1	Measurement Computing	PCI-DDI-08
Analog Input Board	2	National Instruments	E-6023

Table A.6 Schematic reference, part 1

Schematic Reference	Description	Abbrev.	Item Type
A1	Compressor	K	Actuator
A2	Evap 1 Expansion Valve	EEV1	Actuator
A3	Evap 2 Expansion Valve	EEV2	Actuator
A4	Evap 3 Expansion Valve	EEV3	Actuator
A5	Evap 3 Auxiliary Valve	SEV	Auxilary
A6	Condenser Water Pump	WP1	Actuator
A7	Evap 1 Water Pump	WP2	Manual
A8	Evap 2 Water Pump	WP3	Actuator
A9	Evap 3 Water Pump	WP4	Actuator
A10	Zone 1 Disturbance Pump	WP5	Actuator
A11	Zone 2 Disturbance Pump	WP6	Actuator
A12	Zone 3 Disturbance Pump	WP7	Actuator
A13	Water Chiler Pump	WP8	Actuator
M2	Liquid Receiver	LR	Miscellaneous
M3	Filter Drier	FD	Miscellaneous
M4	Condenser Water Chiller	CWC	Miscellaneous
MV01	Compressor Inlet Shutoff	KI	Manual
MV02	Liquid Receiver Inlet Shutoff	LRI	Manual
MV03	Liquid Receiver Bypass Valve	LRB	Manual
MV04	Liquid Receiver Outlet Shutoff	LRO	Manual
MV05	High Pressure Side Access Valve	E1-1	Manual
MV06	Evap 1 Inlet Shutoff (EEV)	E1-I	Manual
MV07	Evap 1 Discharge Shutoff	E1-O	Manual
MV08	Evap 2 Inlet Shutoff (EEV)	E2-I	Manual
MV09	Evap 2 Discharge Shutoff	E2-O	Manual
MV10	Evap 3 Inlet Shutoff (EEV)	E3-I	Manual
MV11	Evap 3 Auxilary Valve Bypass	E3-B	Manual
MV12	Evap 3 Discharge Shutoff	E3-O	Manual
MV13	Low Pressure Access (Suction/Fill)	K-1	Manual

Table A.7 Schematic reference, part 2

Schematic Reference	Description	Abbrev.	Item Type
MWV01	Zone 1 Pump Supply	MWV1	Water Flow
MWV02	Zone 2 Pump Supply	MWV2	Water Flow
MWV03	Zone 3 Pump Supply	MWV3	Water Flow
MWV04	Condsenser Pump Supply	MWV4	Water Flow
MWV05	Zone 1 Disturbance Choke	MWV5	Water Flow
MWV06	Zone 2 Disturbance Choke	MWV6	Water Flow
MWV07	Zone 3 Disturbance Choke	MWV7	Water Flow
S01	Suction Pressure Sensor	PERO1	Transducer
S02	Evap 1 Refrigerant Mass Flow	E1	Transducer
S03	Evap 2 Refrigerant Mass Flow	E2	Transducer
S04	Evap 3 Refrigerant Mass Flow	E3	Transducer
S05	Condenser Pressure	PCRO	Transducer
S06	Compressor Current Draw	KCUR	Transducer
T01	Condenser Refrigerant Inlet Temperature	TCRI	Thermocouple
T02	Condenser Refrigerant Outlet Temperature	TCRO	Thermocouple
T03	Evap 1 Refrigerant Inlet Temp.	TERI1	Thermocouple
T04	Evap 1 Refrigerant Outlet Temp.	TERO1	Thermocouple
T05	Evap 2 Refrigerant Inlet Temp.	TERI2	Thermocouple
T06	Evap 2 Refrigerant Outlet Temp.	TERO2	Thermocouple
T07	Evap 3 Refrigerant Inlet Temp.	TERI3	Thermocouple
T08	Evap 3 Refrigerant Outlet Temp.	TERO3	Thermocouple
T09	Condenser Water Inlet Temperature	TCWI	Thermocouple
T10	Condenser Water Outlet Temperature	TCWO	Thermocouple
T11	Evap 1 Water Inlet Temp.	TEWI1	Thermocouple
T12	Evap 1 Water Outlet Temp.	TEWO1	Thermocouple
T13	Evap 2 Water Inlet Temp.	TEWI2	Thermocouple
T14	Evap 2 Water Outlet Temp.	TEWO2	Thermocouple
T15	Evap 3 Water Inlet Temp.	TEWI3	Thermocouple
T16	Evap 3 Water Outlet Temp.	TEWO3	Thermocouple