Adaptive Neuro Energy Management Control Strategies for HVAC Systems in Buildings

Maria Victoria Gómez Weiss

A Thesis

In the

Department
of
Building, Civil and Environmental Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science (Building Engineering) at
Concordia University
Montreal, Québec, Canada

April 2006

Montreal, Québec, Canada, 2006

Your file Votre référence
ISBN: 0-494-14236-7
Our file Notre référence
ISBN: 0-494-14236-7

ABSTRACT

Adaptive Neuro Energy Management Control Strategies
for HVAC Systems in Buildings

Maria Victoria Gómez Weiss

Energy consumption in buildings is either directly or indirectly related to HVAC systems. As buildings increase in size their corresponding cooling loads also increase, which leads to the necessity of having evermore efficient cooling systems. Control systems and energy management strategies play an important role in improving the overall performance of the system, also allowing the reduction of energy consumption with existing equipment. However the design and implementation of these strategies is not trivial. It is the purpose of this study to develop and implement a series of neural networks (NNs) for energy management control strategies (EMCS) in a model. These Neuro EMCS are: start and stop lead times, temperature base economy cycle, supply air and water temperature reset, and gain selector for the PI controller. To determine the potential benefits and possible energy savings a comparison is made between the developed NNs and a previously developed and tested EMCS algorithms (Base Case). The results show that the adaptive NNs perform very well and as such they are considered as good candidates for implementation in real building systems.

ACKNOWLEDGEMENTS

I would like to give my appreciation to Professor Dr. Mohammed Zaheer-uddin, for all the patience and time that he dedicated in the realization of this project, for the enriching conversations that we had, and the valuable guidance through out my entire masters. Also I would like to thank him for understanding the difficult times that I went through, and for his economical support.

I would like to thank Professor Nicolai Tudoroiu for his valuable assistance in the use of neural networks and for his help in the first stages of this project.

I thank my family, Javier, Victoria, Mauricio and Vicky for believing in me, and for always keeping my spirits high.

Special thanks to Ernest, whom encouraged me, gave me invaluable counsel during the writing process, and also for his optimistic views.

Also thanks to Stephanie for her invaluable company and help, and to Dimitri for his guidance in the GUI programming.

To Ants, Louise and my volleyball team for cheering me and understanding my time needs.

To all of you, whom I failed to mention and helped me, I am really grateful.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Symbol	Variable Represented
A	Face area of the coil
A_f	Area of the fins
Afo	Ratio of the fin area to the total area
Ai	Internal area of the tube
Ait	Internal area of the tube per unit of length
Ao	Total heat transfer area air side per unit of length
A_t	Area of the tube
Cfin	Specific heat of the fin
COP	Coefficient of performance of the system
*COP*max	Maximum coefficient of performance
Cpa	Average specific heat constant pressure for the air
Cpw	Average specific heat for water
Ct	Specific heat of the tube
Cv	Average specific heat constant volume for the air
d	External diameter of the tube
Dh	Hydraulic diameter
din	Internal diameter of the tube
finthick	Fin thickness
hit	Heat transfer coefficient water side
ho	Heat transfer coefficient air side
ht	Heat transfer coefficient coil – air
j	Design parameter
JP	Design parameter
k_a	Thermal conductivity of air
k_{fin}	Thermal conductivity of the fin
k_w	Thermal conductivity of water
Lc	Length of the coil
Ma	Mass flux of air

Symbol	Variable Represented
Ma_1	Mass flux of air for zone 1
Ma_2	Mass flux of air for zone 2
Maw	Ratio of mass flux of air
mfin	Mass of the fins per unit of length
mt	Mass per unit of length
Mw	Mass flux of water
Mwa	Ratio of mass flux of water
p	Neural Network input
Pr_a	Prandlt air
Pr_w	Prandlt water
q_{s1h}	Maximum zone 1 cooling load
q_{s1l}	Minimum zone 1 cooling load
q_{s2h}	Maximum zone 2 cooling load
q_{s2l}	Minimum zone 2 cooling load
qz	Cooling load for the zone
qz_1	Cooling load for zone 1
qz_2	Cooling load for zone2
Re	Reynolds of water
Re *din*	Reynolds of the internal diameter
St	Stanton number
t	Actual time
t	Neural Network Targets
$T\infty,t$	Temperature of the mechanical room
Ta	Mean air temperature
Ta,in	Temperature entering the coil
Th	High temperature
t_h	Maximum temperature occurrence time
Tl	Low temperature
t_l	Minimum temperature occurrence time
To	Outside temperature
Tsa	Temperature of supply air
Tt	Temperature of the tube

Symbol	Variable Represented
Tw	Mean water temperature
Twr	Temperature of water return
Tws	Temperature of water supply
Tz	Zone temperature
Tz_1	Temperature of the zone 1
Tz_2	Temperature of the zone 2
$Tzfin$	Expected final temperature of the zone
$\Delta Tmax$	Delta of temperature maximum for the cooling tower
Uhp	Power input for the heat pump
Uhpmax	Maximum power input
Uo	Overall heat transfer coefficient
$Vface$	Velocity of air
$Vtank$	Volume of the storage tank
V_w	Water velocity
Vz	Total volume of the zones
Vz_1	Volume of zone 1
Vz_2	Volume of zone 2
x_a	Fin height
x_b	Fin width
xv	Percentage of outside air in the building

GREEK LETTERS

Letter	Variable Represented
α	Ratio between the total heat transfer area and the total volume of the coil
γ	Specific heat ratio for air
η_o	Overall efficiency
η_s	Fin efficiency
μ_w	Water viscosity
ρ_w	Density of water
σ	Ratio of total face area to total area of the coil

ABBREVIATIONS

Abbreviation	Meaning
adNN	Adaptive Neural Network
Ave.	Average Error
BC	Base Case
EMCS	Energy Management Control Strategies
EMS	Energy Management Strategies
HVAC	Heating, Ventilation and Air Conditioning
ki	Integral Gain in a PI Controller
kp	Proportional Gain in a PI Controller
MIMO	Multiple Input, Multiple Output
Max.	Maximum Error
mwnn	Gain Selector Neural Network for the Supply Air Temperature Loop
NN	Neural Network
PCA	Principal Component Analysis
RLS	Recursive Least Squares
SISO	Single Input, Single Output
SLTNN	Start Lead Time Neural Network
STPNN	Stop Lead Time Neural Network
tsann	Supply Air Temperature Neural Network
TsasetNN	Set Point Generated by the Neural Network
TwsetNN	Set Point Generated by the Neural Network
twsnn	Supply Water Temperature Neural Network
Tz1nn	Gain Selector Neural Network for the Zone 1 Temperature Loop
Tz2nn	Gain Selector Neural Network for the Zone 2 Temperature Loop
uhpnn	Gain Selector Neural Network for the Supply Water Temperature Loop
VAV	Variable Air Volume
xvnn	Outdoor Air Intake Neural Network

1 INTRODUCTION

1.1 Introduction

Modern way of life is contributing to tremendous strain on our natural resources. As the time passes, this trend is growing rapidly. As a consequence, energy bills for buildings both industrial and commercial, in general have become a very significant part of their budget. This, combined with the general concern with other associated problems like pollution, and the lack of alternative clean sources to supply actual power needs, is the motivation for the development of strategies to diminish the energy consumption at all levels. To this end, heating, ventilating and air conditioning (HVAC) systems in buildings are the ideal candidates to focus for reducing the energy consumption without compromising comfort and productivity.

The literature shows the generalized effort to reduce energy use. For example, energy providers study peak hours to produce the power needed at all times of the day and not over produce power (Wenzenberg and Dewe, 1995) while research is also conducted about how, where and at which hour power should be used to save energy, aiming to prevent energy interruptions and overloads, as well as reducing pollution.

Numerous studies exist in developing strategies to manage energy consumption of HVAC systems in buildings. These strategies vary greatly in complexity and in subject of study, for instance, neural networks for fault detection or energy management strategies (EMS) like stop and start lead times, heating/cooling load prediction or logical rules to predict the behaviour and proper response of the system (Fuzzy Rules). Yet, the main objective is the same: to reduce energy bills as well as make equipment maintenance and

repair cheaper and more infrequent. An accepted and widely used method for testing those strategies is by way of simulation and modeling of building's dynamics (Mathews and Botha, 2003). This method will be used in this thesis to combine several EMC (Energy management control) functions in a two zone, variable air volume (VAV) HVAC system. It is recognized that the EMC strategies are dependent on the cooling or heating loads of the building, which result in complexities in developing efficient algorithms for tuning of PI gain constants. The control performance deteriorates due to ongoing dynamic changes of the system. In response to this need this research work attempts to use the ability of neural networks (NNs) to generalize and adapt a set of Neuro - EMS. The Neuro – EMS are developed in two stages, a non adaptive and an adaptive group. Both will be implemented under the same conditions and compared with Base Case EMCS (Huang, 2003), to evaluate the performance and energy reductions obtained by the different systems.

1.2 Objectives

This thesis focuses in the development of neural networks for EMCS. The following are the objectives:

1. Develop of a two zone VAV HVAC system model with four PI control loops for the implementation of the different EMCS and Neuro EMCS.
2. Implement and simulate previously developed EMCS algorithms (Huang, 2003) in the VAV HVAC system model to generate the Base Case results. This Base Case is the point of reference in the energy consumption comparison between different strategies. The EMC functions include:
    - Start Lead Time

- Stop Lead Time
- Temperature based economy cycle: Proportional Band
- Supply air and water temperature reset.

3. Develop and train a set of non adaptive neural networks for the individual control loops, using data from the Base Case simulations. Integrate the individual Neuro PI controllers and the Neuro EMCS, in order to compare the energy consumption of this system with that of the Base Case.
4. Create and train online a set of adaptive neural networks for building an adaptive Neuro EMCS, not dependent on the data generated by the Base Case algorithm.
5. Compare the three developed EMCS in terms of their performance and energy consumption and suggest the best possible system according to the advantages and capabilities of each system.

1.3 Thesis Organization

A literature review (Chapter 2) discuses previous work in the field of neural networks related to EMCS and to PI controller gains. It is noted that there is lack of research that attempt to combine more than two EMCS, or PI gain controller selection. A model is established using the previously tested EMC algorithms described by Huang (2003), generating what we call a Base Case (Chapter 3). Then in Chapter 4 the group of non adaptive neural networks (NNs) are developed to replace the EMC algorithms and to select the PI gains for the controllers. These two models will be compared in Chapter 5 with the second group of adaptive NNs, developed in the same chapter. The differences, improvements, advantages and disadvantages of each of the above mentioned strategies are discussed.

2 LITERATURE REVIEW

2.1 Introduction

Condition monitoring of HVAC systems plays an essential role in energy management systems. Sophisticated control strategies are becoming easily available; these strategies when implemented are expected to improve energy efficiency and comfort. The strategies are beginning to be used worldwide and currently integrate different versions of several diagnostic and control tools. Development of a new generation of EMS is underway and is expected to provide additional system enhancements and combine the functions of the existing tools. The new EMCS include the use of artificial intelligence to automate, improve the quality of the analysis, provide timely alerts, and the use of an Internet link for collaboration, remote operation and supervision. These enhancements are designed to provide an intelligent system, which can do more of the prediction and decision making, while continuing to support the depth of analysis currently available at existing operations. This research work presents a development of a computational strategy in energy management systems (EMS) that include the application of neural networks. To this end, an efficient and effective global data need and control strategy must first be defined, then an appropriate set of neural networks selected and specified to accommodate that need. This must be performed to maximize the efficiency of the EMS and overall performance of the system.

Consequently, in this section some of the research on the fields of modeling, EMCS algorithms and neural networks correlated with control and prediction in HVAC systems was reviewed.

2.2 Modeling and EMCS Algorithms

A series of research papers on the growing field of EMS and energy efficiency will help to abreast this important technology. Strategies for Energy Efficient Plants and Intelligent Buildings form a comprehensive knowledge base. These strategies can be used to reduce operating costs and improve overall efficiency of any facility as well as maintain desired levels of comfort for the users.

In order to generate data that is adequate for training the set of neural networks, it is necessary to develop a model. Since the early 1980s there have been a lot of contributions in this field. Roberts and Oak (1991) developed a model for a single zone heating system based on energy and mass conservation principles. The model was subject of comparison with data taken from the real system. The results were in general conformed with the measured data.

Zaheeruddin and Zheng (1994) developed a series of coupled equations using the basic principles of energy, momentum and mass balance in a VAV system for dynamic modeling and control. Later on (2001), the same authors used a simplification of these equations to implement a multistage optimal control strategy, based on the building's schedule, the utility rates according to the time of the day, the energy storage and the building loads. Making a comparison between three different cases: constant volume (CV), variable air volume with constant set points (VAV) and with variable set points (VAVN), they were able to show optimal operation for the three cases, leading to savings of up to 25% in the VAVN case compared with the CV case.

Further on, Huang (2003) developed and compared the performance of several EMC strategies. The author proposed a set of improved algorithms. According to the compared

energy consumption with a base case, savings over 10% in energy consumption were reported.

Virk, et al. (1994), used a stochastic multivariable technique to develop their model. The main purpose of the research was to create a predictor for the humidity and temperature inside an unfurnished zone. Using experimental data from a test zone they were able to validate the ability of the model to cope with system changes and to predict short and long term thermal and moisture behaviour. They suggested for further investigation that in the modeling process a moisture absorbent material be included in the zone, in order to include the effects as function of time. Further studies were made by Virk et al. (1994b) by taking a building zone subject to natural environmental conditions. This was compared to the already developed model with the addition of the energy management predictive On/Off strategy. Further comparison was made between the standard On/Off strategy with a hysteresis of ± 1C and the model based On/Off strategy, obtaining a superior regulation translated in 17% extra energy savings. However, the authors noted that the model does not account for the number of occupants or the heat gains related to open doors and other appliances, and as such the results would vary when these effects are included.

House and Smith (1996), used the governing equations derived from mass and energy in a two zone building HVAC system to compare three types of control strategies: conventional, optimal and near optimal. The authors conclude that although the greatest cost savings are achieved by means of optimal control, this strategy does not take into account the comfort of the occupants and the energy efficiency that is desired. They also

state that the near optimal control strategy is easier to implement in real systems, with satisfactory results.

Yoshida et al.(2002) centre their study on an optimal operation scheme for existing HVAC systems in buildings. First they establish the mathematical models that will represent the existing components and their energy consumption. This includes effects of humidity changes in the environment. However no heat losses from pipes or ducts are taken into account. Then they determined optimal set points based on energy consumption of pumps, fans and the heat pump, while satisfying the cooling demand and considering that the system operation and loads are in steady state. The modeling results predicted up to 30% in savings while the real implementation, although successful only rendered 13% energy savings. These results were measured on days when the cooling load was small.

Semsar et al. (2003) also designed a nonlinear controller, but in this case for a non linear model of a MIMO HVAC system. The nonlinear model is combined with the non linear controller to obtain a linear system, then by the back stepping method the new system feedback was designed. The simulations show good decoupling of the disturbances and output regulation.

Haves et al. (2001) discuss the difficulties in creating accurate models that can be used to quantify and compare the energy consumption of an actual building with the design values for the same. Furthermore the related issues of communications between control systems inside a building are evaluated with respect to feasibility and performance of EMCS.

Arguello and Velez (1995) included in the modeling of their HVAC system a thermal load estimator. The authors created a bilinear model for simulating the HVAC plant dynamics, with the intention of improving the efficiency of the controller designed. Since the design of the controller requires all states to be known, a reduced order observer to estimate the thermal loads acting upon the system and the heat exchanger temperature was developed. Also a Lyapunov – based disturbance rejector is designed and coupled with the controller, showing better performance then the regulator without it. The results obtained for the cost performance showed that the disturbance rejector is not only more efficient at maintaining comfort conditions but also shows lower energy consumption in the water pump.

2.3 PID Control in HVAC Systems

It is recognized that the most common way of controlling HVAC systems is by using a PID controller. This is due to the relative low cost and easy tuning of this type of controller. However the high nonlinearity of the HVAC systems and the increased consciousness about energy consumption has rendered the necessity to improve or replace this type of controller.

Geng and Geary (1993) generated an improved Ziegler-Nichols set of rules for the design and selection of a PID controller for an HVAC system. Using load disturbances simulated by means of a first order equation with transport delays, the authors were able to study the effects of these disturbances on the overshoot and settling time of the closed loop responses of the system. During some of the simulations Geang and Geary only use the proportional gain to observe the range of operation of the system. They noted that the Ziegler–Nichols tuning based on open loop parameters is valid for a small time delay,

and that these are less conservative than those based on ultimate sensitivity based parameters.

Anderson et al. (2002), worked developing and testing a MIMO controller for an experimental facility. After arranging in the canonical form the plant models a robust PI control strategy is developed. Using data proportioned by the experimental system the main variables (Mass flow rate of air, mass flow rate of water, temperature of the supply water and temperature of the supply air) and the out door air temperature as a disturbance, the authors compare three controller structures with the corresponding PI controller. A minimal MIMO robust controller, a constrained MIMO robust controller and a full MIMO robust controller showed improved results in the settling times and disturbance rejection. Even though the best results were obtained with a design that has independent water circuits for each of the air handling units, improvements in the settling times were achieved with robust MIMO controllers that share the supply water with other air handling units.

Nesler (1986) generated and tested in a seven zone HVAC system an adaptive PI controller. The strategy includes a parameter estimation step (Recursive Least Squares), a design calculation step, an automatic tuning and a performance monitor, which are connected to both the PI controller and the process. To avoid abrupt changes in the outputs while the change in the PI gain constants is made, the controller is modified to provide bumpless changes in the process. One interesting aspect presented by this author was the rules to avoid parameter estimation while expecting great changes in the system, avoiding unnecessary computation and possible miss tuning of the controller. The results

obtained from the testing of the algorithm were satisfactory; nevertheless RLS algorithm was limited by unmodeled process disturbances and the actuator hysteresis.

Wang, et al. (1999) developed and tested under experimental conditions an advanced PID auto-tuner for both SISO and MIMO processes. For this purpose the process critical point is determined experimentally by relay tests and later presented as a transfer function. The number of iterative processes to find the transfer functions is proportional to the number of variables that are used in the system. The system implemented in Java is accessible via internet and allows three types of operation: PID control, Manual and Auto tuning, ensuring a bumpless transition between the auto-tuning and the conventional PID with manual control. Hailin and Broberg (2002) also developed Internet based HVAC applications for remote control. Using an internet link up to all the control systems in the building, monitoring and energy management strategies such as scheduling are easy to manage from a distant location.

Wang, et al. (2001) developed a PID auto tuner. The authors assumed a second order plus dead time model that represented the HVAC system dynamics. They used the relay tests developed by Wang et al. (1999) to provide the constants to validate the model and designed the PID controller. The strategy was tested in a HVAC pilot plant for the room pressure loop and the supply air pressure loop. Improved results were obtained by using the PID tuning rules.

He and Asada (2003) proposed a linear model for the evaporator, and a non linear controller based on this model. By combining several of these models, the system can simulate multiple zones. The authors state that for a wide range of operation the PI controller will not need auto tuning due to the non linear compensation included in the

controller. It is also mentioned that just by removing the equations corresponding to the i^{th} zone, the controller can change the input to the system allowing it to cope with on/off EMS. The simulation showed that even with large estimation errors the control can still produce the desired results.

2.4 Neural Networks and Control

The growing importance of HVAC systems and the need for improving load management, economy and energy costs is a major feature of several references. HVAC design techniques including the latest advances in equipment and controls are thoroughly covered. Currently the research in control strategies using neural networks is really vast. Countless researchers have explored the possibilities to enhance the performance of controllers with the aid of predictive techniques by artificial neural networks (ANN). However to the best of our knowledge, the work done to implement EMCS in HVAC systems is considerably less. In this part of the literature review, we intent to check the most relevant studies for this research project.

To begin, Miller and Seem (1991) developed a three layer feed forward neural network with two input nodes (the zone temperature and the outdoor air temperature) and the start or stop lead time as a result. Using a building simulation program, data for four cities was generated, with different building materials and sizes. A comparison between the different developed architectures (different number of neurons in the hidden layer, different learning rates, initialization techniques, etc.) and a recursive least square algorithm shown below:

$$\tau = a_0 + (1-w)(a_1 T_{room,initial}) + w\, a_2 T_{ambient}$$

where

$$w = -\left(\frac{(T_{room,initial} - T_{set,night})}{(T_{room,final} - T_{set,night})}\right)$$

$T_{room,initial}$ = *room temperature at the beginning of the return period.*

$T_{room,final}$ = *room temperature at the end of the return period.*

$T_{set,night}$ = *setpoint temperature during night or weekend setback.*

τ = *return time.*

a_0, a_1, a_2 = *coefficients determined with recursive least squares.*

The results showed by the authors in terms of maximum error, standard deviation and average error, showed almost no difference between the two methods. The reasoning behind this similarity of performance was justified by the lack of diverse data in the 90 days of simulation. Nevertheless the level of expertise, computational power and time required for the artificial network to work was higher for the ANN, rendering it a bit impractical to implement. The authors also developed a new training strategy that improved the training times of the network in 90% from the traditional training algorithms.

Tudoroiu and Zaheeruddin (2004) developed a neural network for PID tuning for the discharge air temperature control loop. After comparing the performance and computational time of several MIMO and SISO functional blocks, the authors showed that the standard deviation and the mean error on the MIMO-Neuro model were significantly lower, however the SISO model give good enough predictions at a smaller computational cost. The neural networks selected was a three layer feed forward neural network, with the control loop error as the input, the differential and integral gains as the hidden layer outputs and the estimated discharge air temperature and proportional gain as targets. The data sets for the offline training were obtained with experimental data and

the NN retrains online to cope with disturbances during the simulation in the discharge air temperature control loop. This architecture reduced the oscillations in the actuator and gave good tracking performance.

Fargus and Chapman (1998) developed a hybrid PI-Neuro controller, considering the HVAC system dynamic behaviour as a first order system. Using a closed loop proportional controller combined with a feed forward NN that acts as a distributed integrator, the authors showed that in approximately two days the network is capable of online development of an inverse model to predict and provide a robust control. Since the network was implemented based on a weighted radial basis function, with the basis function uniformly distributed on a rectangular grid, it was recognized the limitation of functioning of the network in the active domain, forcing constant changing between the normal PI controller and the hybrid one. The simulation results showed that online training is possible by stretching the active domain and after checking the data obtained from the network for at least five independent periods of time the network can take over the normal PI bumplessly.

Pingkang et al. (2002) developed a neural network control algorithm for replacing a PID controller with a Genetic Algorithm integrated to optimize the learning rates for the NN. The results were first simulated and then tested in a robot servomotor of a real HVAC system with a sample time of 15 seconds. The neural network PID-like controller showed robust control without the need of a model or parameter tuning.

Semsar et al. (2003) designed a control methodology where the main objective is to reduce actuator repositioning. Combining the neural network with a conventionally

designed PID controller the authors show that the use of the non linear element to chop the oscillations of the actuator, to give better tracking performance and energy reduction. Saboksayr, et al. (1995) designed a robust decentralized controller for a multizone space heating to compare with a decentralized NN controller. A neural network produced optimal results when the sensitivity of the cost function was included in each of the corresponding feed back gains. The results showed that the NN performed better than the robust decentralized controller, and that the best performance was observed while on training mode. Nevertheless, the recall mode and the adaptive mode showed acceptable results (Tudoroiu and Zaheeruddin, 2004; Roberts and Oak, 1991; Huang, 2003).

Ahmed et al. (1998, Part I, II and III) used a different approach taking advantage of the properties of the general regression neural networks (GRNN) to capture the dynamics of the cooling coil of a real HVAC system. They created a feed forward control strategy that had satisfactory results when compared with the standard PID controller. Part of the success was due to a supervisory controller that they implemented in order to ensure smooth transitions whenever the adaptive system made the changes in both the pressure and temperature control sequences. Nonetheless is good to note that in a real system both controllers will have to interact at the same time, while in the simulations they were tested independently.

To continue with the system identification and prediction, Chow and Teeter (1997), used the conservation of energy and mass principles to develop an ANN model. Using a set of 5 normalized inputs, they compared their results by measuring the magnitude of the input layer weights. This way of selecting the neural network called functional link, allows the ANN to achieve excellent generalization properties while eliminating meaningless

operation in the input layers. The same authors used their developed model (Zaheeruddin and Zheng, 1994) in an attempt to minimize a given cost index, using the ability of this particular type of neural network to determine the relevant inputs in the system while still following the dynamic changes in the system. It is worth to note that the results obtained were based on a single zone system.

Song et al. (2003) developed a model of a VAV system using an adaptive neural network, mainly concentrating in the air flow and motor-fan dynamics for VAV control. Since the behaviour inherent in these systems present high uncertainty and non linearity, the implementation of an adaptive mechanism for the ANN with robust stability and convergence was presented. Lower tracking errors were obtained with the robust tuning algorithm compared to those from the standard back propagation, showing that this strategy can deal with model uncertainty and system disturbances.

Curtiss et al.(1993) implemented two neural networks, one predictor one controller, based on the experimental data provided by a real building. The research was subdivided in local and global control of the HVAC system in a commercial building. For the local control a comparison between the conventional PID controller response and the PID-Neuro controller was made, showing stability and better time responses in the temperature reset control EMCS. For the global control scheme, they concentrated in the plant optimization and energy savings by prediction in heat loads when ANN predictor was used in adapting the set points to the experienced loads.

The same authors (1994) using data again from a real system generated an optimized plant operation through regression modeling and an ANN plant optimizer control. After comparing the results by both methods, it was corroborated that the use of ANN is

feasible for the system modeling and even better than the regression modeling. Then four EMC methods were installed for comparison: Fixed set points, reset set points, ANN optimizer, and ANN optimizer with duct pressure reset. The results showed that the worst case scenario for energy consumption was related to the reset set points strategy while the best was achieved by the ANN optimizer with duct pressure reset showing savings up to 15% compared to the standard fixed set point. The authors recommend further studies to take into account the peak load, since the ANN optimizer tends to keep the water supply temperature in the lowest possible through out the normal operation day.

Wezenberg and Dewe (1995) developed two adaptive recursive neural networks, for forecasting the local tariff rates due to a plan to constantly change them depending on local needs for energy. This project was developed as an effort to efficiently supply the needs of hot water in a house based on the forecasted consumption for the next 30 minutes at the lowest rate possible. Using a hybrid 3 layer/2 layer neural network to predict the amount of hot water required in the next three hours the system was able to match the available profiles and determine the heating periods for the house hot water cylinder with 336 data points for initial training. The authors state that this sort of networks can be easily implemented in commercial buildings.

Daryainian and Norford (1994) developed a discrete time model of a house, and created an online optimization procedure to match the requirements of energy of the building taking into account the variable rates for the peak energy consumption. Although the model is limited (SISO, linear and time invariant), the relative facility of implementation is really appealing while savings up to 19% in energy consumption were obtained. Of

course these results were also influenced by the size of the storage tanks, and external factors, as outside temperature and occupancy while the experiments were accomplished.

Henze and Hindman (2003) developed clustering neural networks to control an air cooled chiller - condenser. Comparing task-blind and task-specific training, the authors found that the error from the task-blind is considerably higher; nevertheless, the training of these networks can be unsupervised and relatively fast. The authors conclude in the paper by suggesting the use of the task-blind neural network in the beginning and allowing the clustering layers to train under task-specific conditions to improve the results.

Hepworth and Dexter (1994) proposed a neural control scheme that compensates for the plant non-linear behaviour and degradation via an adaptive algorithm. The simulation includes the set point set back EMC, with two schedules, to compare the performance of the normal PID controller and the NN. Three inputs with 40 basis functions (radial basis functions) were used as the architecture for these experiments. The results showed that the neural control scheme have better overall results than those from the standard PID controller, however careful choosing of the radial basis functions have to be made in order to avoid estimation inaccuracies that can lead the neural network to higher errors.

Other useful papers for this research are referenced here.

Alessandri,et al. (1994), evaluated several network architectures in order to obtain a balance between computational cost and precision of the predictions. The idea was to implement an optimal control strategy in the conditioning of a green house heating system, aided by the prediction of the outside air temperature. The results yielded that the use of a simple ARMA (12, 2) was enough to forecast the outside temperatures and the dynamic effects on the green house. The authors also advised that the training is more

effective when the data is generated from a mathematical dynamic model than from the "black box" approach.

Parlos et al. (2002) use NN to approximate the unknown dynamics of for an adaptive state filtering. Creating separate NN models for each filter mapping, the authors obtained acceptable results when Kalaman filters and Extended Kalaman Filters fail to converge. Due to the inaccurate predictions of the network in the beginning, a two stage offline training is used to provide an effective overall training. The NN provide an alternative to constant measurement of some state parameters in the system that would be too expensive to keep in constant track.

In respect of principal component analysis (PCA), Erguo and Jinshou (2002) presented a non linear PCA for input training NN for fault detection, with the purpose of showing that the linear PCA can omit valuable information. This is because of the lower portion in the variability of the system from some of the nonlinear variables. Using data from a continuous stir tank reactor, and a back propagation NN combined with the input training network to estimate the non linear principal component scores online.

Confronting the computational inability to manage large data sets for the training of neural networks Colmenares and Perez (1999), created a method to reduce the number of observations and variables without compromising the reliability of the data set. The authors used stratification on the original dataset and using samples from each stratum to produce a new and reduced data set. Comparing the results obtained training the NN with the data given by the stratification method, the stratification method combined with PCA and the complete data set, it was shown that the stratified/PCA method yielded the best results.

Li, Vaezi-Nejad and Visier (1996) created a multilayer feed forward neural network for fault detection. Using typical errors encountered in heating systems, the authors created a discrimination tool to aid the operators to determine the source of degradation in the system. The data was obtained by simulating the system under different conditions, faulty and not faulty and later on tested with an unseen data set. The results demonstrated the feasibility of using neural networks for fault detection and diagnosis, however it also exposes the necessity to have a network that does not required retraining for each heating system that is going to be applied to.

2.5 Fuzzy EMC and Control Systems

As mentioned before, artificial intelligence methods are beginning to be widely used for improving the operation and performance of HVAC systems. It is necessary then to review other methods than NN. In this case Fuzzy EMC and Control systems were evaluated.

Egilegor, et al. (1997), combined a neural network with a fuzzy control system to provide HVAC control of humidity and temperature in a three zone building. Simulation runs are performed with different values for the fuzzy control parameters, and a neural network is trained to select an optimal between these values to satisfy the HVAC system. The PID control gain values are selected by means of linguistic variables in the Fuzzy control, taking into account the humidity levels in the zones. With this data set the neural network is trained offline to select the fuzzy variables. Different climate changes were tested offline showing improvements mostly in the winter weather. However the implementations required for the online training and testing have not been developed yet, so there were no results available.

Hawkings et al. (1990) used the ability of the fuzzy control rules to be combined, to achieve a better control strategy for the outdoor air dampers.

It is expected to have better control in the zone temperature, by just improving the performance in the supply air ducts. Further experiments were conducted in a full scale building facility by Dexter et al.(1990) . The authors implemented jacketing rules for the PID controller in order to supervise the behaviour when transport delays are significant. The fuzzy rules were implemented and tested with the supply air temperature produced by an air handling unit obtaining good results when compared to those of a self tuning PI controller.

Cooper and Warwick (1994) use a two level fuzzy logic controller to produce expert control in different systems and environments. The second level of control uses three mini controllers to minimize the overshoot, the rising time and the plant usage (turn on and off of the cooling plant). Combining these by means of fuzzy rules, the overall performance of the system is improved, while the first level is in charge of reaching operational temperatures after the night setback. Changes in the plant in extended periods of time take into account by the adaptive algorithm, however since the original rule base is kept, they infer that the future adaptations are expected to decrease the chances of instability.

Pargfrieder and Jorgl (2002) generated an integrated system to control the HVAC system along with the blinds and artificial illumination by means of fuzzy control. Using ANN predictions of the external conditions of the zone are given. Records of changes provided by the users, and comfort ratios are used to optimize energy consumption without compromising the human comfort. A MIMO system with partial cross coupling is used to

adapt the fuzzy inputs to the heat profile, blind positioning and solar radiation of the system. An optimization algorithm runs after some time of operation, with the stored data from the different data channels to reduce the energy required by the system.

Using the generalized predictive control algorithm combined with the optimization of the energy input to the system is showed that the comfort of the occupants can be improved, while still achieving energy consumption savings.

Kuntze and Bernard (1998) responding to the increase demand in real systems to react to occupancy and event responsive systems, created fuzzy controller for the system for ventilation control and temperature control. Taking into account the user requirement by means of a "comfort economy slider", the system is capable of determining the adequate energy consumption for the desired control levels of the occupants. Choosing the zone temperature, relative humidity and the levels of CO_2 as control parameters, the authors created a quasi static model with only two parameters for optimization. Creating a fuzzy model of the comfort criteria outlined in the ASHRAE "Ventilation for acceptable indoor air quality" (1982), and a membership function describing the required heating power, the reference temperature is statically optimized. The Air exchange rate is optimized by means of a dynamic feedback optimization, using a predictive model to relate the relative humidity and the CO_2 levels. The two optimal values are combined by the economy factor provided by the user. The authors conclude that this strategy can be combined with other EMS, and that further study is needed to be able to include this system in commercial buildings, rather than in homes.

Huaguang and Cai (2002) constructed a decentralized non linear adaptive controller to cope with different load conditions in a HVAC system. Dividing the system in a fuzzy

logic inner loop and in a Fourier integral based control outer loop, the authors ensure that the system output vector tracks the desired trajectory, while guaranteeing a zero steady state error.

Lea et al. (1996), also took into account the relative humidity in the zone as well as the desired temperature and evaluated the performance of their fuzzy control strategy in a test House. The data collected from the system was used to evaluate the performance, for tuning and further design. The multiple zones are controlled by four variables, the compressor speed, the fan speed, the required mode (heating, cooling or neither) and the damper position. Three Fuzzy variables were used with five fuzzy decision making rules. The results showed good comfort control, temperature wise and humidity wise, for the six zones. However no energy consumption comparisons were made so further studies and testing is required.

Kiff and Warwick (1996) studied the interactions between different zones, in a group of offices. Using the microprocessors of relative low power that are distributed in the nodes in each room, the processor deals with fuzzified data, and deals with the defuzzification locally, avoiding having one central system dealing with all the fuzzy data, and permitting future enhancements in newly added nodes. Three fuzzy control variables are chosen at each node, with trapezoidal and triangular functions to ease calculations. Two layers of fuzzy associative memory are used to close the control loop, and fuzzy singletons are used to defuzzify the data and give it as a percentage with a sample time of 5 minutes. However no results of the effect of the controller have been reported, since the controller is still in trial stage. Additional information transmitted from the rooms to a

centralized controller would allow better decisions in set point selection and heat input when having different zones interacting.

Arima et al. (1995) also showed temperature and humidity control, by means of fuzzy logic controller. In this case, the HVAC system uses two fuzzy logical variables, temperature and humidity, and control them by means of the hot water valve, the damper positioning and the mass flow rate of water in the humidifier. A set of seven fuzzy rules with triangular functions is decided for the humidity control while seven fuzzy rules were used for temperature control. The two sets of rules are combined and applied in an experimental site composed of 4 rooms, showing better performance for both humidity and temperature control, when compared to the rough set method, providing well control environment. However further improvement can be achieved when trying the system with VAV.

Rahmati, et al. (2003), create a hybrid fuzzy and PID controller for a MIMO VAV-HVAC system. Using Sugeno's fuzzy rules, generalized forms of Ifs and THENs were employed to control both the temperature and the humidity of the zone. It was shown that the Fuzzy PID controller had better tracking speed and robustness than that showed by the regular PID.

Ying-guo et al. (1998) compared the results obtained for a single zone SISO system with a PID controller, a fuzzy PID and adaptive Fuzzy controller. The research showed that the adaptive control method based on fuzzy sliding mode control, is able to cope with the frequent changes and disturbances in the zone, using a simplified fuzzy control, handling the fact that some linguistic rules are difficult to collect in some situations. The simulations based on a swimming room showed satisfactory results, while exhibiting

simplicity and applicability. Hamidi and Lachivier (1998) approach the indoor control problem by means of a fuzzy control system that maintains a constant indoor air comfort instead of a constant indoor air temperature. The control parameters being part of human preferences are subjective and imprecise, presenting a perfect opportunity to implement a fuzzy controller. Using six inputs regulated by means of seven fuzzy rules, the HVAC system receives the air velocity, the set point for the supply air temperature, and the expected thermal comfort level in the zone. Although the simulation results show positive results and simplification of otherwise a long iterative process, the implementation of such system in real buildings would require sensors for data acquisition for air velocity, relative humidity and interactive panels to receive feedback from the users. Angelov et al. (2000) created a methodology for modeling HVAC components by using an encoding mechanism that evolves the fuzzy rules base structure and parameters from the training data. The use of just a few rules to control generated by means of NN and Genetic Algorithms, the system gives the possibility to follow up easily the system outputs, as compared with other black box approaches for modeling components. The authors also emphasise in the ability of the system to acquire expert knowledge, during creation and use of the model, while decreasing the computational requirement to obtain the outputs from the model. Jian and Wenjian (2000) compared the results obtained by a normal PID controller, a Fuzzy logic controller and an adaptive neuro-fuzzy controller in regulating the supply air pressure. Using three/six Takagi and Sugeno basic fuzzy rules adjusted by a five layer adaptive neural network, the system is able to update the rules to current conditions while still keeping the last set of rules. The system performs comparably to a well tuned PID system, however the experiment shows a limited capacity to deal with the

steady state error. The authors solved this limitation by adding a second integrator loop that can be switched on or off according to the change in the error with the set point. Furthermore the controller shows its strength when dealing with conditions farther from the design point. Mathews and Botha (2003) used simulation as a tool for improved thermal building management. Taking data from a real building to corroborate the developed model good results for the predicted zone temperature and the energy consumption were obtained. By including system degradation and change in conditions, the authors are able to detect the fouling and investigate alternative control strategies that would allow the real system to operate relatively efficient while the HVAC system is back in to proper operation. The authors conclude from their study that this type of simulation tools can aid the operation and maintenance departments to enhance scheduling and general energy management in the building. Wen and Smith (2001) studied the effect of the thermostat time constant in the energy consumption. Comparing four thermostats with different time constants in an interior zone of a VAV HVAC system, the authors were able to show that although the difference in time constant does not alter the stability of the temperature control, a slow thermostat causes higher energy consumption and slight discomfort to the occupants in the zone.

From this literature review we can conclude that many efforts have been made in improving the performance of controllers without compromising the occupant's comfort. However, these efforts seem to be separated one from another, or having just one or two EMC functions combined at a time. We intend in this work to include at least four Neuro – EMC functions combined and evaluate the potential energy savings obtained by their combination by means of comparing it with a base case model with the EMC algorithms.

3 DEVELOPMENT AND IMPLEMENTATION OF THE PLANT MODEL AND OF THE ENERGY MANAGEMENT CONTROL SYSTEM (EMCS): BASE CASE

3.1 Introduction

The objective of this Chapter is to describe the components of the Base case. 'Base Case' refers here to the two – zone variable air volume HAVC system model (i.e. plant) with selected EMC strategies (Huang, 2003). Firstly, the equations that comprise the three components or submodels of the plant are presented. These submodels consist of the storage tank dynamics (i.e. water supply temperature), followed by the cooling coil model (which is composed by the return water temperature, the tube temperature and the supply air temperature), and the zone temperature model. Since the cooling coil forms the interface between the zone models and the storage tank dynamics a section in this chapter was dedicated to summarize the design and constant selection of the cooling coil. Secondly, the implementation of the plant (all three submodels) in Simulink ® (Mathworks, 2002) is done. The results obtained from the implementation are divided in three main categories: open loop simulations, closed loop simulations, and finally the Base Case, in which the EMC strategies are implemented. Lastly, the conclusions related to this chapter are presented.

3.2 VAV HVAC System Model Components

In this subsection a model to simulate the dynamics of a two zone VAV, HVAC systems is described. A set of differential equations based on energy balance principles, are used

to describe the dynamics of the four independent control loops. This plant was used to implement Energy Management Control (EMC) functions developed by Huang, 2003. In the following part the physical model and the model equations are described. All variables related to the equations used in this work can be found in Appendix 1.

3.2.1 VAV System Layout:

Figure 3.1 depicts the physical layout of the VAV, HVAC system that comprises the plant model, and its correspondent control loops, based on two office rooms.

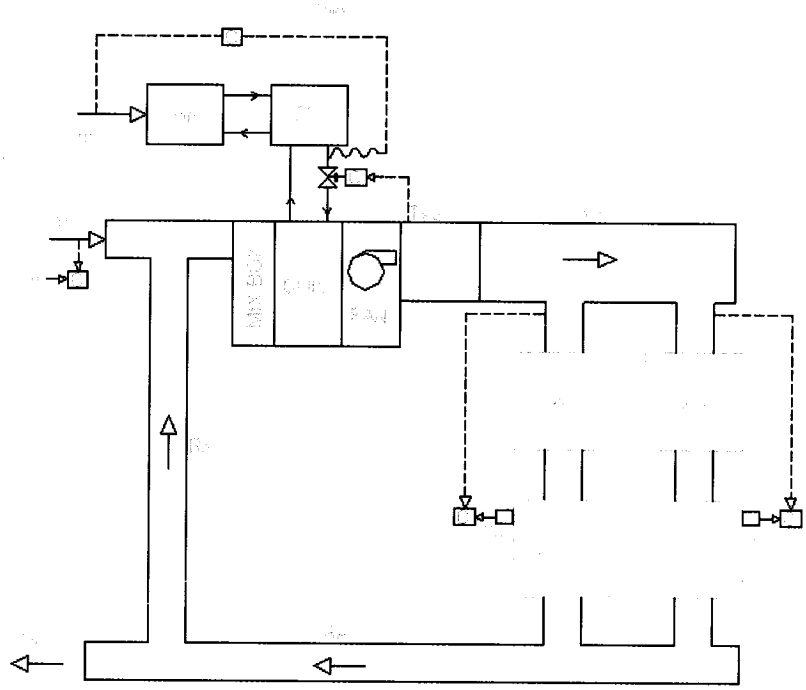


Figure 3.1. System's Layout (Hung, 2003)

The system is composed of two zones (Z_1 and Z_2), each having independent temperature set points (*Tz1set, Tz2set*). Both zones are 3.3m in length, 3m in width and 2.8m in height. They both have one external wall facing south. The rest of the walls are considered to be interior. Z_2 is considered to have a higher cooling load than Z_1. The system was designed

to provide two refrigeration tons of cooling, using a VAV HVAC system. The heat pump (HP) provides refrigeration to the storage tank (ST) where the chilled water that supplies the cooling coil is stored. The air from the mixing box is cooled in the coil and then circulated to the zones. The supply air temperature (SA) is modulated by means of regulating the mass flow rate of water that goes through the coil. The Variable air volume boxes (VAV) are used to supply appropriate mass flow rate of air to each zone in order to maintain their respective set points. The air that returns (RA) is partially exhausted (EA) and partially mixed with outdoor air (OA) in the mixing box, completing in this way the air circulation cycle.

3.2.2 Water Storage Tank Model:

The variable of interest in this submodel is the supply water temperature, *Tws*. The water that is supplied to the cooling coil, is provided from a tank of volume *Vtank*, density, ρw, and specific heat, *Cpw*. An energy balance on the chilled water tank yields Equation 1:

$$\frac{dTws}{dt} = \frac{1}{\rho w \cdot Cpw \cdot Vtank}\left\{-Mw \cdot Cpw(Tws - Twr) - Uhp \cdot Uhp\max \cdot COP + \alpha_h(T\infty, t - Tws)\right\} \tag{1}$$

The energy extracted by the heat pump is determined by the multiplication of the percentage of power input, *Uhp*, times the maximum power input for the heat pump, *Uhp*max. The coefficient of performance (COP) of the heat pump is modeled according to Equation 2:

$$COP = (COP\max)\left(1 - \frac{To - Tws}{\Delta T\max}\right) \tag{2}$$

Where COPmax is the design coefficient of performance of the system and ΔT max represents the maximum difference of temperature between the water and the air in the cooling tower. *To* is the outside temperature and it is modeled with a cosine profile. Since all the submodels will depend on the predicted outside air temperature profile, its model equations and a sample profile will be described in detail below.

3.2.2.1 Outdoor Air Temperature Profile

The outdoor air temperature profile is dependent on the expected maximum (T_h) and minimum (T_l) temperature values, and their corresponding time of occurrence (Cho,S. and Zaheeruddin, M., 2003). If the time of the day is less than the time of the lowest temperature occurrence then the outside temperature is modeled by Equation 3.

$$To = Tv - Td.\cos\left[\left(\frac{\pi}{24-(t_h - t_l)}\right)(t - t_l)\right] \qquad when \quad t \le t_l \tag{3}$$

If the time of the day is in between the times of occurrence of the lowest and highest temperatures expected on that day, then Equation 4 is to be used.

$$To = Tv - Td.\cos\left[\left(\frac{\pi}{(t_h - t_l)}\right)(t - t_l)\right] \qquad when \qquad t_l \le t \le t_h \tag{4}$$

Finally if the time of the day passes the time of the highest temperature, then Equation 5 shown below will be applicable.

$$To = Tv - Td.\cos\left[\left(\frac{\pi}{24-(t_h - t_l)}\right)(t - t_h)\right] \qquad when \qquad t_h \le t \tag{5}$$

Where,
$$Tv = \frac{T_h + T_l}{2}, \qquad Td = \frac{T_h - T_l}{2}, \tag{6}$$

T_h and T_l are the maximum and minimum outdoor temperatures respectively for the 24 – hour period; t is the current time, t_h is the time of occurrence of the maximum temperature, and t_l is the time of occurrence of he minimum temperature.

Although most of the simulations shown in this work are made for summer conditions the same equations were used for winter, autumn and spring season by matching minimum and maximum temperatures to produce the appropriate temperature profiles.

3.2.3 Cooling Coil Model:

The cooling coil is of the typical counter – cross flow type with continuous plate fins on its tubes. The model equations are adapted from (Huang, 2003). The first equation describes the water return temperature *Twr*, the second equation describes the tube temperature *Tt*, and the third equation describes the supply air temperature *Tsa*:

$$\frac{dTwr}{dt} = \frac{hit \cdot Ait}{mw \cdot Cw}\left(Tt - \bar{T}w\right) + \frac{Mw}{mwLc}\left(Tws - Twr\right) \tag{7}$$

$$\frac{dTt}{dt} = \frac{1-\eta s}{\eta s + \dfrac{mt \cdot Ct}{mfin \cdot Cfin}}\left\{\frac{Cpa \cdot Ma}{\rho \cdot A \cdot Lc}\left(Tsa - Ta,in\right) - \frac{hit \cdot Ait}{mfin \cdot Cfin\left(1-\eta s\right)}\left(Tt - \bar{T}w\right) + \left(\bar{T}a - Tt\right)\left(\frac{\eta so \cdot hc \cdot Ao}{\rho \cdot Cv \cdot A} + \frac{\eta so \cdot hc \cdot Ao}{mfin \cdot Cfin\left(1-\eta s\right)}\right)\right\} \tag{8}$$

$$\frac{dTsa}{dt} = -\frac{ht \cdot \eta o \cdot Ao}{\rho \cdot Cv \cdot A}\left(\bar{T}a - Tt\right) - \frac{\gamma \cdot Ma}{\rho \cdot A \cdot Lc}\left(Tsa - Ta,in\right) \qquad \gamma = \frac{Cp}{Cv} \tag{9}$$

Tt, *Tw* and *Ta* represent mean bulk temperatures of the tube, the water and the air within the coil. The reason why these temperatures are considered as 'mean bulk' is to avoid the

transient effects of the change in the temperature along the length of the tube. *Ta* and *Tw* are described in Equation 12 and Equation 13.

$$Maw = \frac{Ma \cdot Cpa}{Ma \cdot Cpa + Mw \cdot Cpw} \tag{10}$$

$$Mwa = \frac{Mw \cdot Cpw}{Ma \cdot Cpa + Mw \cdot Cpw} \tag{11}$$

$$\bar{T}a = Maw \cdot Ta,in + Mwa \cdot Tsa \tag{12}$$

$$\bar{T}w = Maw \cdot Twr + Mwa \cdot Tws \tag{13}$$

These equations do not take into account the effect of conduction between the tube, water and air of the coil due to the small contribution of this effect as compared to the convection effect, and also because it simplifies the model and allows for faster calculations.

The temperature at the entrance of the coil (Ta,in) results from the mixture of the outdoor air temperature (*To*) and the return air temperature (*Tz)*. The variable *xv* refers to the proportion of outdoor air that is admitted in to the building.

$$Ta,in = xv \cdot To + (1 - xv) \cdot Tz \tag{14}$$

The heat transfer coefficients (*hit, ht, hc)*, the overall efficiencies used for these set of equations (ηo, ηs, ηso) as well as the dimensions and properties of the coil (*Ait, Ao, A, Lc, mfin, Cfin, mt, Ct) were* calculated using the methods outlined in reference (McQuiston, 2000); a summary of these calculations is presented in section 3.2.5.

3.2.4 Zone Model:

A two zone model was developed. An energy balance on both zones yields the following equation:

$$\frac{dTz}{dt} = \frac{1}{\rho a \cdot Cpa \cdot Vz}\{Ma \cdot Cpa(Tsa - Tz) + qs + \alpha z(To - Tz)\} \quad (15)$$

Since the two zones have the same volume and similar characteristics (but different loads), the same equation can be used to describe both zones, with the appropriate variables to accommodate the characteristics of each zone, as shown by Equations 16 and 17.

$$\frac{dTz_1}{dt} = \frac{1}{\rho a \cdot Cpa \cdot Vz_1}\{Ma_1 \cdot Cpa(Tsa - Tz_1) + qs_1 \cdot \alpha z(To - Tz_1)\} \quad (16)$$

$$\frac{dTz_2}{dt} = \frac{1}{\rho a \cdot Cpa \cdot Vz_2}\{Ma_2 \cdot Cpa(Tsa - Tz_2) + qs_2 \cdot \alpha z(To - Tz_2)\} \quad (17)$$

For estimating the cooling loads acting on each zone, profiles similar to those for the outside temperature were created, again based on the maximum and minimum expected loads and their corresponding times of occurrence, as depicted in Equations 18, 19 and 20.

$$qs_i = qv - qd.\cos\left[\left(\frac{\pi}{24 - (t_h - t_l)}\right)(t - t_l)\right] \quad for \quad t \le t_l \quad (18)$$

$$qs_i = qv - qd.\cos\left[\left(\frac{\pi}{(t_h - t_l)}\right)(t - t_l)\right] \quad for \quad t_l \le t \le t_h \quad (19)$$

$$qs_i = qv - qd.\cos\left[\left(\frac{\pi}{24 - (t_h - t_l)}\right)(t - t_h)\right] \quad for \quad t_h \le t \quad (20)$$

Where,

$$qv = \frac{q_h + q_l}{2}, \qquad qd = \frac{q_h - q_l}{2}, \tag{21}$$

q_h and q_l are the maximum and minimum cooling loads respectively during the 24h – period. The parameters used in all of the equations are summarized in Appendix 1.

3.2.5 Cooling Coil Design:

In order to implement the model of the VAV HVAC system it is necessary to design a cooling coil that will meet the specified cooling load. To satisfy this requirement a cooling coil was designed using the steady state techniques described by McQuiston (2000). The first step is to determine the cooling capacity of the coil. According to the design cooling loads of the zone, the peak load could be satisfied with 1.5 tons of refrigeration (5280W). Therefore a coil capable of sustaining 2 Tons of refrigeration was selected. The following mass flow rate of air was found to match the cooling load of the 2 Tons.

madot		
qload	7040	W
Delta T	12	K
madot	0.584	kg/s

Table 3-1. Design Air Mass Flow Rate

A Delta T of 12 degrees is used since the desired set point for the zones was chosen to be 25 C and the temperature of the supply air to the zones was set at 13 C. Based on these assumptions, the average values of the heat transfer coefficients inside the tube (*hit*), and in the surface in contact with the air (*ho*), were computed using the recommended values for velocities of the air and water (McQuiston, 2000). The results of these calculations are depicted in *Table 3-2*.

hit		
Vw	1	m/s
Redin	9531.25	
hit	3137.345	W/ m^2.K
ho		
Sigma	0.54	
Va	4.57	m/s
Gfr	5.49	kg/m^2.s
Gc	10.19	kg/m^2.s
Red	9008.03	
Dh	0.0026	m
A/At	14.47	
JP	0.0175	
J	0.0068	Fig. 14-14 (McQuiston, 2000)
ho	88.073	

Table 3-2. Coil Design Heat Transfer Coefficients

Once these values were calculated, the efficiencies, ηs (fins) and ηo (overall), were determined (*Table 3-3*) for a four row triangular fin configuration:

ηs		
dim1	0.015875	m
dim2	0.01588015	m
l	0.01588015	m
m	0.015875	m
Psi	1.99685535	
Beta	1.00032443	
Re/r	2.12226672	
Fi	1.41783772	
m	75.2712334	
Ns	0.81355515	
ηo		
Afo	0.942	Afin/A
No	0.82436895	

Table 3-3. Coil Design, Overall Efficiency of Fins and System

The overall heat transfer coefficient *Uo* and the effectiveness of the system were then computed and listed in *Table 3-4* and *Table 3-5*. The size and specifications of the coil are summarized in *Table 3-6*.

Uo		
Alpha	800.0	M
Ai/Vc	43.88375689	1/m
Ai/Ao	0.054854696	
1/hono	0.013709129	m^2.K/W
1/hit(Ai/Ao)	0.005810648	m^2.K/W
Uo	51.23009135	W/ m^2.K

Table 3-4. Coil Design, Overall Heat Transfer Coefficient

NTU		
Tws	7	C
Twr	11	C
Tain	25	C
Tsa	13	C
Ca	591.59	W/K
Cw	1760	W/K
Cmin/Cmax	0.336	
Epsilon	0.6648	
NTU	1.34	Fig. 14-18 (McQuiston, 2000)

Table 3-5. Coil Design, Number of Transfer Units

Coil Dimensions		
Ao	16.52	m^2
Vc	0.0207	m^3
Vol air/s	0.4865	m^3/s
Afr	0.1073	m^2
Wc	0.1925	m
Hc	0.127	m
Lc	0.8448	m
Nr	6.575	7
N tubes per row		
mwdot	0.45	kg/s
Aw	0.00045	m^2
Ntr	3.826	4

Table 3-6. Coil Design, Dimensions of the Coil and Mass Flow Rate of Water

The first design was made with the assumption of 4 rows of tubes. A second iteration with 7 rows of tubes followed. Both results were very close with only minor changes in the design values. The coil design parameters in each case were very similar. These parameters are summarized in *Table 3-7*:

x_a	0.03175	m	ρ_{tube}	8522	Kg/m^3
x_b	0.0275082	m	*Ait*	1.073	m^2/m
D	0.0159	m	*Mfin*	3.68	Kg/m
Din	0.0122	m	*Mt*	38.97	Kg/m
Cpa	1005	J/kg.K	*Ao*	20.24	m^2/m
ρw	1000	kg/m^3	*A*	0.107	m^2
Cpw	4189	J/kg.K	*Lc*	0.845	m
Kfin	204	W/m.K	*Wc*	0.1925	m
Finthick	0.0001524	m	*Hc*	0.127	m
fin pitch	472.44	1/m	*Ntr*	4	
ρ_{fin}	2707	kg/m^3	*Nr*	7	

Table 3-7. Coil Design, Chosen Parameters for the Equations

In order to simulate the operation of VAV HVAC systems with the designed coil, several properties of air and water are required (e.g. density, conductivity). These were modeled as functions of temperature. The functions were determined by fitting polynomial equations to the property data by Incropera (1996), and these equations are listed in *Table 3-8*.

The equations for water are valid for temperatures between 0C and 92C. Those for air are valid over a temperature range of -73C to 127C.

Property	**Equation**	**R^2**
Water Viscosity	$\mu w = -0.0025 \cdot Tws^3 + 0.5496 \cdot Tws^2 - 44.673 \cdot Tws + 1711.8$	0.9988
Water Conductivity	$kw = -0.0072 \cdot Tws^2 + 1.84 \cdot Tws + 569.29$	0.9991
Air density	$\rho_a = 355.12 \cdot Ta,in^{-1.0033}$	1
Air Viscosity	$\mu a = 2 \cdot 10^{-7} \cdot Ta,in^{0.7967}$	0.9996
Air Conductivity	$ka = 0.0002 \cdot Tai^{0.8986}$	0.9997
Jp vs. j4	$j4 = 0.2737 \cdot JP + 0.0014$	0.9988

Table 3-8. Functions for Variable Parameters of Air and Water

3.3 Model Implementation Using SIMULINK®

To facilitate the implementation of EMC functions, it was convenient to simulate the EMC algorithms in Simulink ® (Matlab). In this manner we will be able to couple the different elements of EMC system in an easier manner. By selecting the start and end dates as input arguments to the algorithm, it is possible to simulate several days of the plant operation, and to continuously monitor the outputs of the system. The use of this particular program environment might also facilitate the use of real data or real time implementation of the program.

3.3.1 Open Loop Simulation:

The VAV system model equations are known to be stiff. The first step taken in order to implement and visualize the response of the model was the open loop simulation. These open loop responses were obtained using the design values for the mass flow rates of water and air in each zone and only 50% of the maximum capacity in the coil (*Uhp = 0.5*). The results of this first simulation run (open loop) for the Zone temperatures, compared to the outside temperature, are depicted in *Figure 3.2*. Since the main disturbances in the system are *To* and the cooling loads, and those have cosine profiles it is expected that the open loop responses of the simulated temperatures also display the same behaviour.

The open loop responses of the supply and return water temperatures and are shown in *Figure 3.3*, while the tube and supply air temperatures are depicted in *Figure 3.4*.

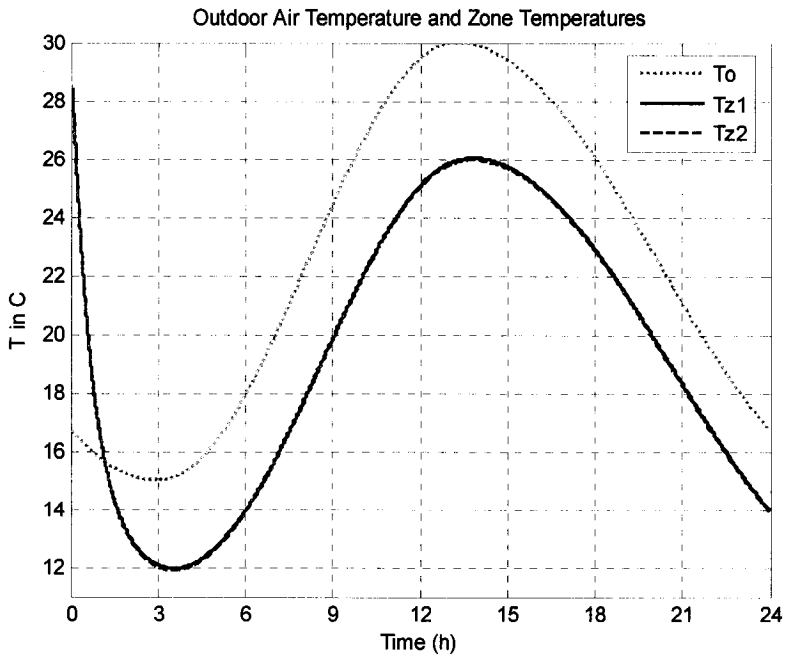


Figure 3.2. Open Loop Outdoor Air and Zone Temperatures

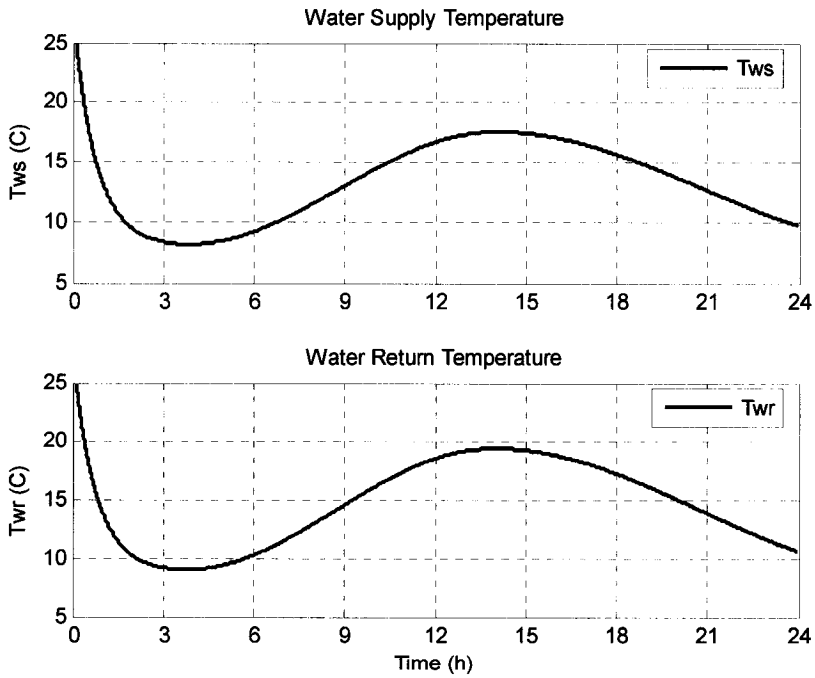


Figure 3.3. Open Loop Water Supply (Tws) and Return Water (Twr) Temperatures

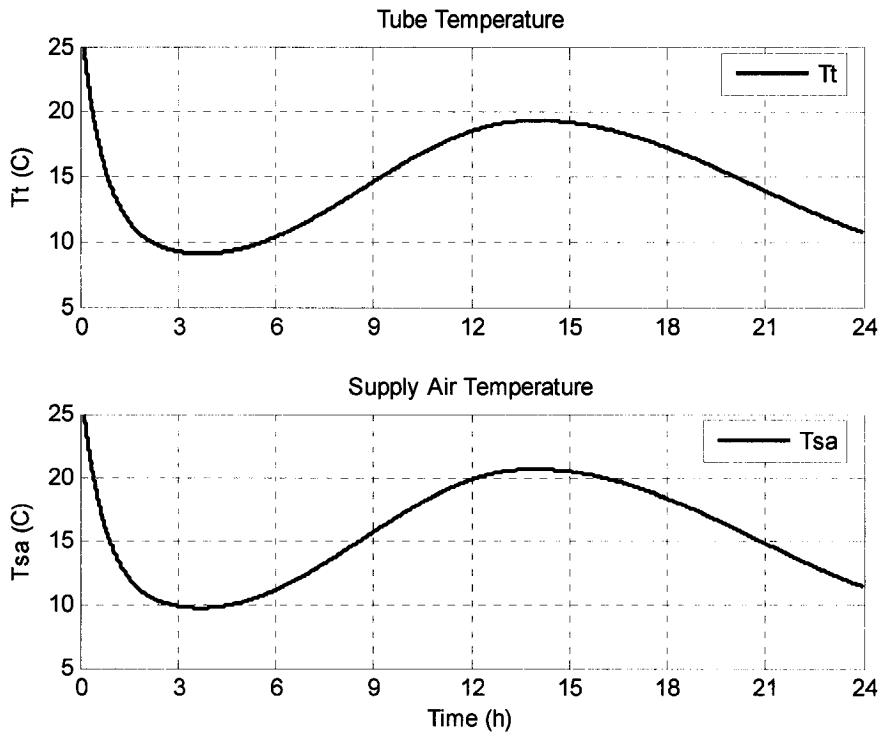


Figure 3.4. Open Loop Temperature of theTtube (Tt) and Temperature of Air Supply (Tsa)

3.3.2 Closed Loop Simulation:

In the two zone VAV system, the following control loops were simulated: Zone 1 and Zone 2 Temperature control loop, Discharge air temperature control loop, and Chilled water temperature control loop. In *Table 3-9* a list of the control variables chosen for regulating the control outputs is given.

Control variable	**Symbol**	**Sensor**	**Symbol**
Power input to the heat pump	*Uhp*	Supply water temperature	*Tws*
Mass flow rate of water in the coil	*Mw*	Air supply temperature	*Tsa*
Air mass flow rate of zone 1	Ma_1	Zone 1 temperature	*Tz1*
Air mass flow rate of zone 2	Ma_2	Zone 2 temperature	*Tz2*

Table 3-9. Control Variables for the HVAC – System

The PI control was selected due to its wide use in the HVAC system industry. The controller gains were chosen through trial and error. Once the desired response was obtained, simulation runs were made using the following set points: *Twsset* = 7C, *Tsaset* = 13C, Tz_1set = 24C and Tz_2set = 22C.

The selected controller gains are displayed in *Table 3-10.*

	PI	
Variable	**Integral**	**Proportional**
Ma1	0.007	1.2
Ma2	0.008	1.2
Mw	0.0001	0.009
Uhp	0.0009	1.2

Table 3-10. PI Control Gains

A typical set of output response for the supply and the return water temperature are depicted in *Figure 3.5. Figure 3.6* shows the response for the tube (*Tt*) and supply air (*Tsa*) temperatures. It is noticeable that although the supply air temperature set point was reached, the overall control of this variable is not completely satisfactory. Between 12h and 18h, when the greatest divergence of the supply air temperature from its set point is presented (1.3C), the mass flow rate of water reached its maximum, yet the heat pump only works at full capacity while the HVAC – system is in the start up phase (*Figure 3.8*). This explains the reason why the supply air temperature (See *Figure 3.6*) does not maintain its set point and could be corrected by changing the set point of supply water temperature (*Twsset*) according to the requirements of the zones. Furthermore the stiffness inherent to the model allows only a small range of values for *kp* and *ki* for this control loop, affecting the reaction of the controller to sudden changes in the system. The responses for the control variables *Uhp*, *Mw* and *Ma* are depicted in *Figure 3.8.*

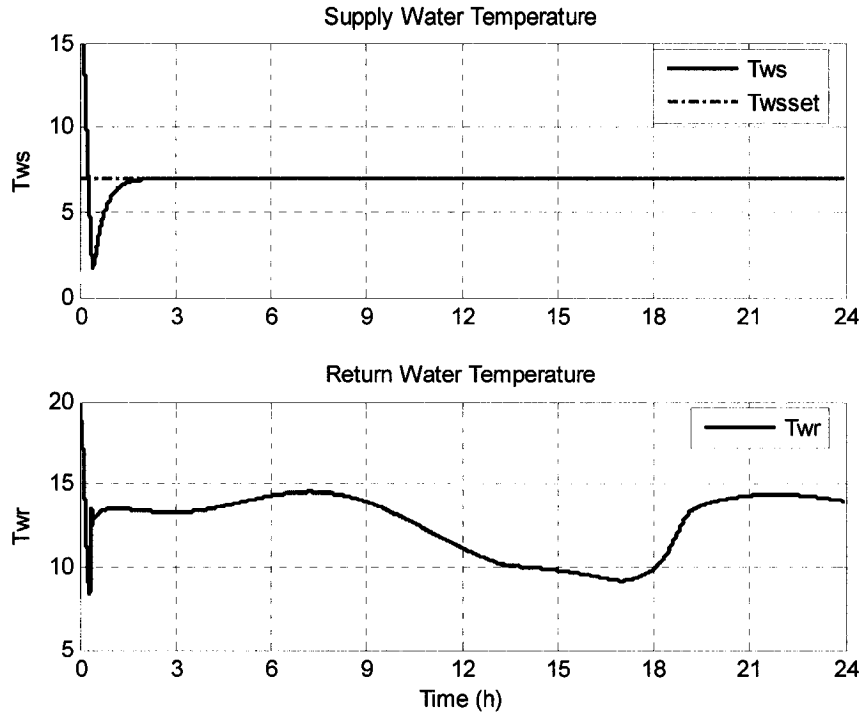


Figure 3.5. Closed Loop Responses for the Water Supply (Tws) and Water Return (Twr) Temperatures

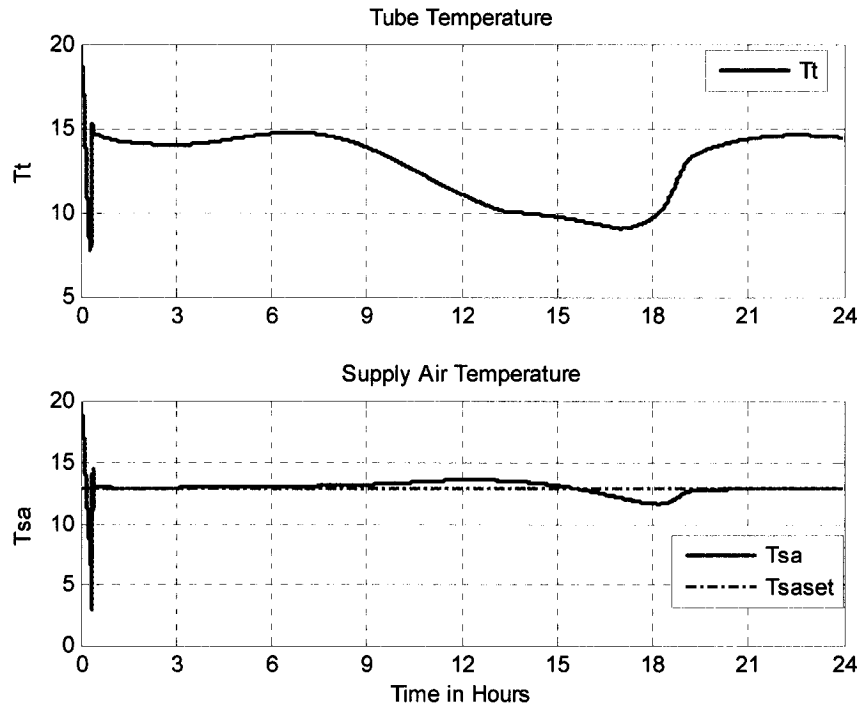


Figure 3.6. Closed Loop Response for the Tube (Tt) and Supply Air (Tsa) Temperatures

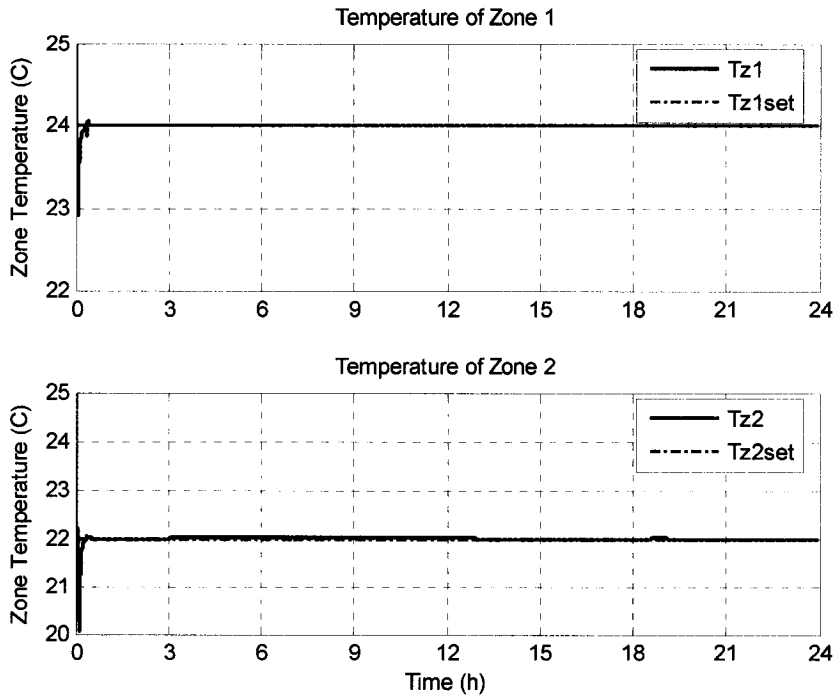


Figure 3.7. Closed Loop Responses for the Temperature of Zone 1 (Tz_1) and Zone 2 (Tz_2)

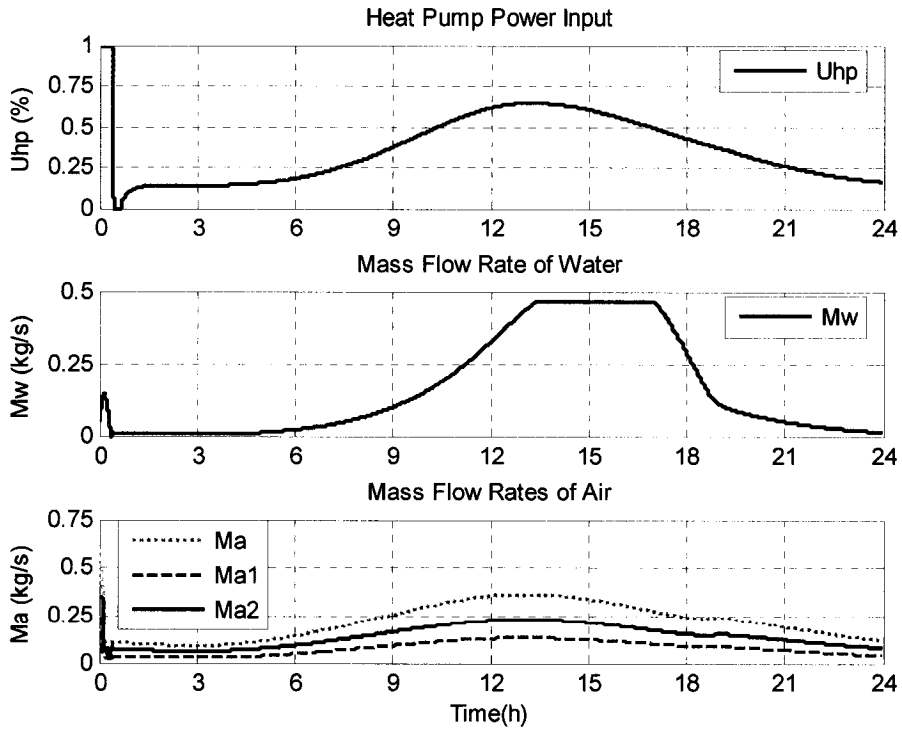


Figure 3.8. Closed Loop Control Variables: Heat Pump Power Input (Uhp), Mass Flow Rate of Water (Mw), Mass Flow Rate of Air (Ma)

In the lower part of the figure (*Figure 3.8*), the individual mass flow of air rates for each zone are depicted as well as the total mass flow rate of air handled by the cooling coil. *Table 3-11* displays the summary of the maximum and average errors for the closed loop simulation. These values are expected to be improved by changing the controller gain values by means of a group of Neuro PI controller gain selectors in later chapters.

	Error	
Variable	**Maximum**	**Average**
Tz_1	1.07	-0.0027
Tz_2	1.93	-0.0045
Tsa	10.00	-0.0015
Tws	5.25	-0.017

Table 3-11. Close Loop Error Summary.

Different initial conditions in the system produce different responses with the same PI controller gains. As a result the response times and overshoot values may vary greatly. From *Figure 3.5* is clear that the longest settling time is that of the supply water temperature control loop. This is due the size of the tank and the greater temperature difference from its initial condition that this system has to attain. Nearly an hour and forty minutes is required for the supply water temperature to reach its set point of 7C. The settling time for the zone temperature is close to 30 minutes (*Figure 3.7*). For these simulations all the temperatures were considered to start at 28C. *Figure 3.9* and *Figure 3.10* shows the settling times for the same model, initializing all the variables with three different temperatures. It is noted that by changing the initial Temperature by just 3C the settling time was reduced to almost a third for the zone temperatures, and the set point of the water supply temperature (*Tws*) can be reached in about an hour.

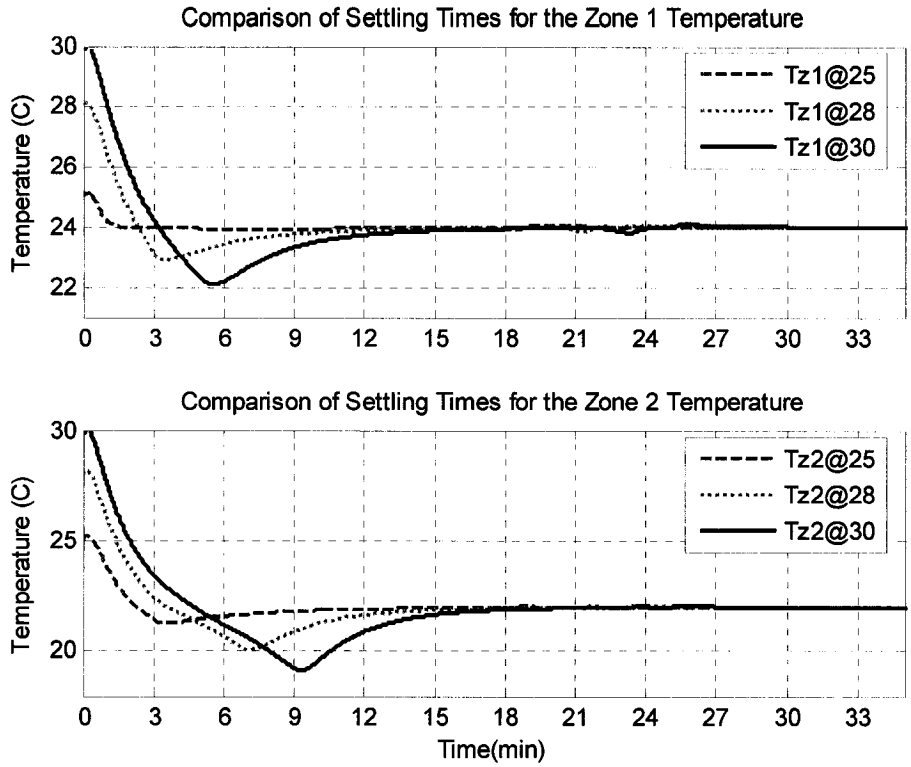


Figure 3.9. Difference in the Settling Time with Other Set on Initial Conditions

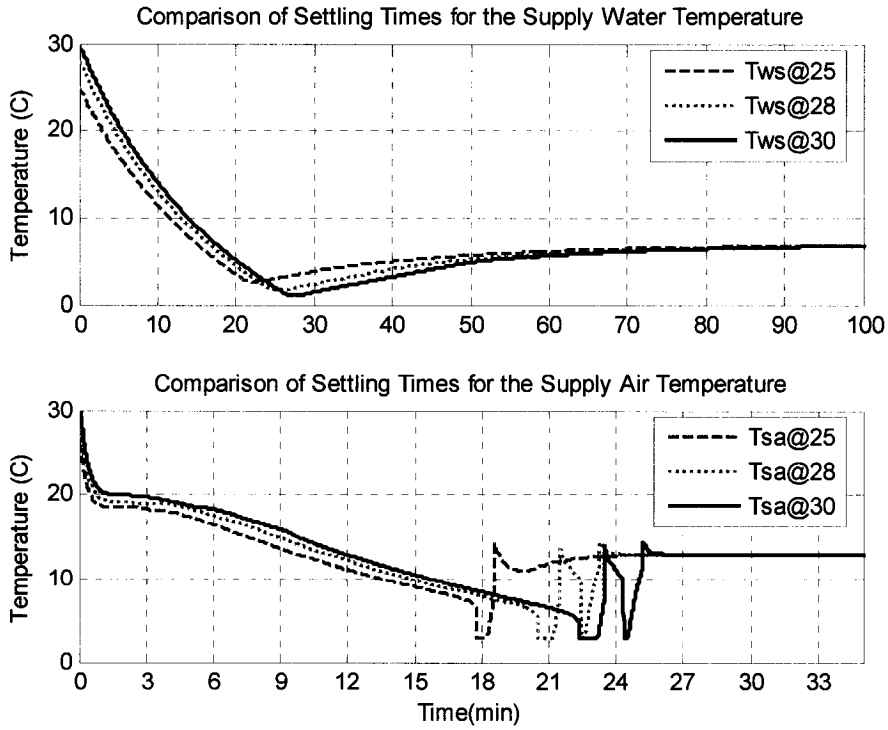


Figure 3.10. Difference in the Settling Time with Other Set on Initial Conditions

3.4 EMCS Implementation

In this section several energy management control functions are implemented in the simulation model discussed in section 3.3. The following EMC functions were considered.

1. Outdoor air Economy Cycle: Proportional Band
2. Supply air and water temperature reset (set point selection via *To*)
3. Seasonal Operating Mode
4. Start and Stop Lead Time Algorithms

3.4.1 Outdoor Air Economy Cycle: Proportional Band

The Proportional band control is implemented to regulate the outside mass flow rate of air admitted in the building. The objective is to take advantage of the cool outdoor air, during the times when its temperature is lower than the temperature of the zone. Also it will restrain the amount of outdoor air admitted into the building to the minimum in the hours that it would produce higher energy consumption. This minimum value was chosen to be $2L/s.m^2$, following the recommendation of Model National Energy Code of Canada for Buildings (MNECCB in Canada, 1997). This value is close to 15% of the total mass flow rate that should be provided in the room (minimum *xv* in the system). The strategy used was the proportional band control, which is described in detail in reference (Hung, 2003), is depicted by Equation 22:

$$x_v = \begin{cases} Tz - n < To < Tz + n; & x_v = x_{v\min} + (1 - x_{v\min})\left(\dfrac{Tz - To}{n}\right) \\ To \le Tz - n; & x_v = 1 \\ To \ge Tz + n; & x_v = x_{v\min} \end{cases} \quad (22)$$

Where n is the proportional band, To is the outdoor air temperature, Tz is the temperature of the zone, considered the same return air temperature (no losses in the ducts are considered), x_{vmin} is the minimum amount of outdoor air that should admitted in the building, and x_v is the actual percentage of outdoor air admitted in the system.

The purpose of the proportional band strategy is to provide a greater amount of outdoor air inside the zones (improving air quality) while decreasing the energy consumption of the HVAC system. This is accomplished by using a linear relation to determine the proportion of outdoor air, when the zone temperature and the outside temperature are close in value (± 2C). If the outside air temperature is lower than the minimum of this range ($Tz - 2$) then only outside air will be brought into the system, providing what is called free cooling. If the outside air is higher than the maximum of the range ($Tz + 2$) then only the required minimum (2L/s.m^2) will be allowed into the building, ensuring that temperature difference between the coil and the air is the lowest without compromising the indoor air quality. This strategy is not used during the winter operation. It is assumed that during this season, the air inside is going to be warmer than the outside air, so admitting more than the minimum in the building would be a waste of energy.

3.4.2 Supply Air and Water Temperature Reset (Set Point Selection Via *To*)

Optimal set point selection is always a good way to economize energy; if the set points correspond to the loads acting on the system and requirements of the users, then the system will run at adequate capacity and the operation cost will be less.

The method implemented in reference (Huang, 2003) was the alternate supply air temperature set point described by Equation (23):

$$T_{saset} = \begin{cases} T_{z,l} < T_z < T_{z,h} & T_{saset} = T_{zset} - \left(\dfrac{q_{s\max}}{M_{a\max} c_{pa}} \right) \left(a_1 T_o^2 + a_2 T_o + a_3 + a_4 \left(T_{zset} - T_{o\_mean} \right) \right) \\ Tz \le T_{z,l}; & T_{saset} = T_{saset,\max} \\ Tz \ge T_{z,h}; & T_{saset} = T_{saset\min} \end{cases} \tag{23}$$

Where T_z, $T_{z,l}$, $T_{z,h}$, are the actual temperature, the lowest temperature and the highest temperature of the zone respectively. T_{zset} is the zone set point. Since there is more than one zone, the temperature chosen for the algorithm is that of the zone with the highest temperature. T_{saset}, $T_{saset,max}$, $T_{saset,min}$ are the actual, the maximum and the minimum supply air temperature set points. q_{smax}, is the maximum sensible cooling load, M_{amax} is the maximum mass flow rate of air in the zones, and c_{pa} is the specific heat of the air. T_o and T_{o_mean} refer to the actual and the expected mean of the outdoor air temperature. a_1, a_2, a_3 and a_4 are coefficients that vary with the peak and the amplitude of the outdoor air temperature. This algorithm selects the set point of air according to the temperature of the zone. The maximum supply air temperature achievable is used when the temperature of the zone is beneath the chosen low limit. When the zone temperature is higher than the chosen upper limit the minimum supply air temperature possible is used. In between, a linear relationship is used, which is based on the zone sensible heat balance equation. The

algorithm depends on the cooling load in the zones and on a prediction of the outdoor air temperature, which leads to difficulties when implementing in real systems (Appendix 2.). A polynomial relationship was implemented between the supply air temperature set point, the temperature of supply water and the actual outdoor air temperature, with the observed performance of the model under different conditions monitored in a variety of 24 – hour simulation runs. The data is included in Appendix 2. *Table 3-12* shows the summary of the corresponding set points while Equations (24a) and (24b) illustrate the functional relationship.

$$Tsaset = -0.2 \cdot To + 17.8746 \quad (24a)$$

$$Twsset = -0.18 \cdot To + 13.6487 \quad (24b)$$

Summer		
To	***Tsaset***	***Twsset***
30	11	7
25	12	8
20	13	9
15	14	10

Table 3-12. Estimation of the Set Points as Function of the Outdoor Temperature

The results obtained from both algorithms were similar. However, the equations based on *To* gave better results in view of the fact that the outdoor air temperature influences most of the dynamic behaviour of the system, and will be an input variable for the purpose of training neural networks. For the energy comparison between these two algorithms, please refer to Figures A1 through A.9 in Appendix 2.

3.4.3 Seasonal Operating Mode

The objective of this algorithm is to select the operating mode of the system based on the expected outside conditions according to the season and date. Using the month of the

simulated date, the algorithm places the operation in between three seasons: summer, winter and spring/autumn combined. The spring and autumn are under one category due the similar need of the buildings during these seasons (cooling, heating or just ventilation). Another important issue in selecting the appropriate operational mode is if the simulated date is a national or local holiday and/or if it is part of the weekend. Knowing this data, the program proceeds to generate suitable profiles for outside temperature and heating or cooling loads for each zone. This allows the simulation to run continuously if required. The algorithm proceeds with the selection of the set points for the zone during unoccupied hours.

For autumn/spring, where heating or cooling might be required, a range of acceptable outdoor air temperatures is generated based on the difference between this temperature and that of the zone. If *To* is in this range then only ventilation is necessary. If the outdoor air temperature surpasses the acceptable range, cooling is required, and the system will interpret the need of the zone as summer. On the other hand if *To* is lower than the acceptable range, then heating will be needed, like in winter. Combining this information with the time of the day and with the knowledge of the simulated date (if it is a holiday, weekend or occupied day) different operating modes are chosen in order to provide the thermal comfort expected from the system. The algorithm also keeps track of the simulated time, with the purpose of providing heating or cooling just during the occupied times, where the loads reach their peak. In this way if the algorithm determines that it is a holiday or a night in the winter (heating is required), a minimum set point of 15C will be kept and no external air will enter. If it is a night in the summer (cooling is required), the heat pump will be shut down, the set point will be kept at 25C and the

outdoor air economy cycle will make use of the *To* which is lower than the zone temperature during the night. Seemingly the outdoor air economy cycle plays an important role during the holidays and weekends since the heat pump is turned off during the length of the days. If the information provided to the algorithm renders that is a spring/autumn holiday the outside temperature is good enough to keep the heating or the cooling systems off and only the ventilation system is needed. The set points will be kept at their last value. During occupied periods in summer or winter, the HVAC system will be operational to reach the set points selected by the users, while in spring/autumn only the outdoor air and the regulation of the mass flow rates of air, would be used in order to reach the desired zone temperatures.

3.4.4 Start and Stop Lead Time Algorithms

Start/stop lead time algorithms are recognized as key elements in the energy savings strategies, due the possibility of diminishing energy consumption during unoccupied hours. But determining the appropriate start or stop lead time is not an easy task; the complexity of the building's distribution and heat and mass transfer, requires that the dynamic behaviour is at some extend studied, measured or known. In the particular case of the algorithm used for the base case, a set of equations and coefficients were developed, to fit the data obtained from the reduce order dynamic model shown below.

$$ETC\frac{dT_z}{dt} = M_a c_{pa}(T_a - T_z) + q_s + \alpha_z(T_o - T_z) \qquad (25)$$

The following equations (26) and (27), shows the algorithm to determine the start and stop lead times used in the program.

$$\begin{aligned} t_{start} &= a_1(Tz - Tzset) + a_2(Tz - Tzset)(To - Tzset) + a_3 \\ t_{stop} &= b_1(Tzfin - Tz) - b_2(Tzfin - Tz)(To - Tzset) \end{aligned} \qquad (26)$$

Where a_i and b_i are coefficients and *Tzfin* is the expected final temperature of the zone:

$$Tzfin = Tzset + 2 \tag{27}$$

These equations and set of coefficients were validated with a more detailed zone thermal model, shown in Equation (28), where C_z is the room thermal Capacity, and the five terms in the right side correspond to the heat exchange in the room, the loads and the infiltration and air changes.

$$C_z \frac{dT_z}{dt} = M_a c_{pa}(T_s - T_z) + \sum_{env} uA(T_{env,j} - T_z) + \sum_{window} uA(T_o - T_z) + q_s + \alpha_z(T_o - T_z) \tag{28}$$

The surface temperature of the enclosure element was modeled with Equation (29), where the heat transfer of the node 'i' with the adjacent nodes 'j' is represented.

$$C_i \frac{dT_i}{dt} = \sum_j \frac{(T_j - T_i)}{R_{i,j}} + q_i \tag{29}$$

A building of medium construction according to the ASHRAE fundamentals Handbook (1997) was chosen for the simulations. However the program can choose the appropriate coefficients for the load and outside conditions simulated, according to the choice of construction introduced by the user (Huang, 2003). Using the information from the seasonal operational mode (previous section), the program is capable of determining the proximity of the occupied/unoccupied periods. Using the settling times from *Figure 3.9* and *Figure 3.10* it is expected than the system will attain the desired set points within an hour. Therefore during weekdays an hour before the expected occupied time (in this case 8am) the algorithm starts calculating based on the available information the possible start lead times. When the simulated time is equal or greater than the start lead time, the system becomes fully operational (winter occupied, summer occupied, spring/autumn occupied). Seemingly, an hour before the expected leaving time (in this case 6pm) the

program calculates the stop lead time. Once the simulated time is equal or greater than the stop lead time the heat pump is turned off, however the water from the storage tank will still be circulated through the coil, providing cooling or heating in the zones. Since in the winter the heat pump is not turned off, even at unoccupied times, the set point is lowered in order to decrease the energy consumption, while taking advantage of the higher temperature of the water in the storage tank.

Table 3-13 depicts the values assigned by the program to each option. The values can be introduced manually in case of need. As explained in the last section during normal weekdays, the night or unoccupied time is considered as a holiday.

Season/Holiday	**Value**
Summer Holiday	1
Winter Holiday	2
Spring/Autumn Holiday	3
Summer Occupied	4
Winter Occupied	5
Spring/Autumn Occupied	6
Summer Stop	7
Winter Stop	8

Table 3-13. Values of the Seasonal Operational Mode: Variable "Logic"

3.4.5 Base Case Simulation Results

The objective of this section is to carryout simulation runs involving all EMC functions described before. This is referred as the base case simulation. The results for the base case run for the zone temperatures are depicted in *Figure 3.11*. As we can see in the figure the zone temperatures converge to their respective set points: Tz1set = 23C and Tz2set = 24C. The highest room temperature between the two zones is the one to determine the percentage of outdoor air in the building, *xv*. From the temperature responses it can be seen that during the unoccupied hours the heat pump is turned off, so

only the fan is working, then during the start time, between 7am and 8 am the heat pump is turned on at the start lead time and the zones are prepared for the occupied period. Between 8am and 5 pm the system runs with the desired set points. Then, during the stop time, we can observe a small jump in the zone temperature, due to the turning off of the heat pump. Afterwards, the night cycle operation takes place.

In *Figure 3.12* is clear that the water supply temperature follows the set points determined in the algorithm explained in section 3.4.2. In *Figure 3.13* the responses for the tube and supply air temperatures are depicted.

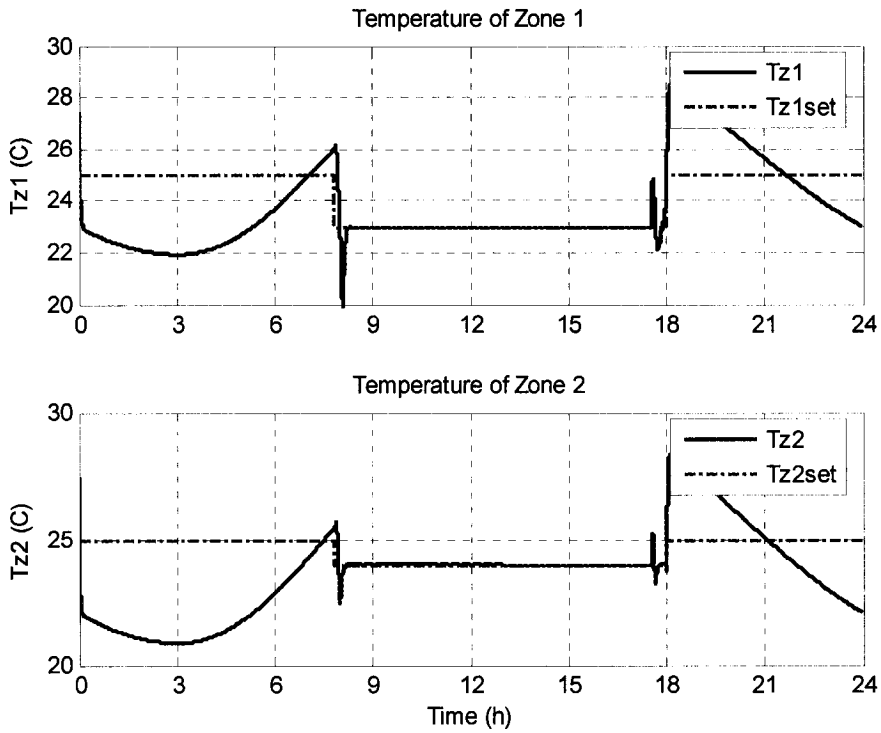


Figure 3.11. Base Case Responses for Tz1 and Tz2

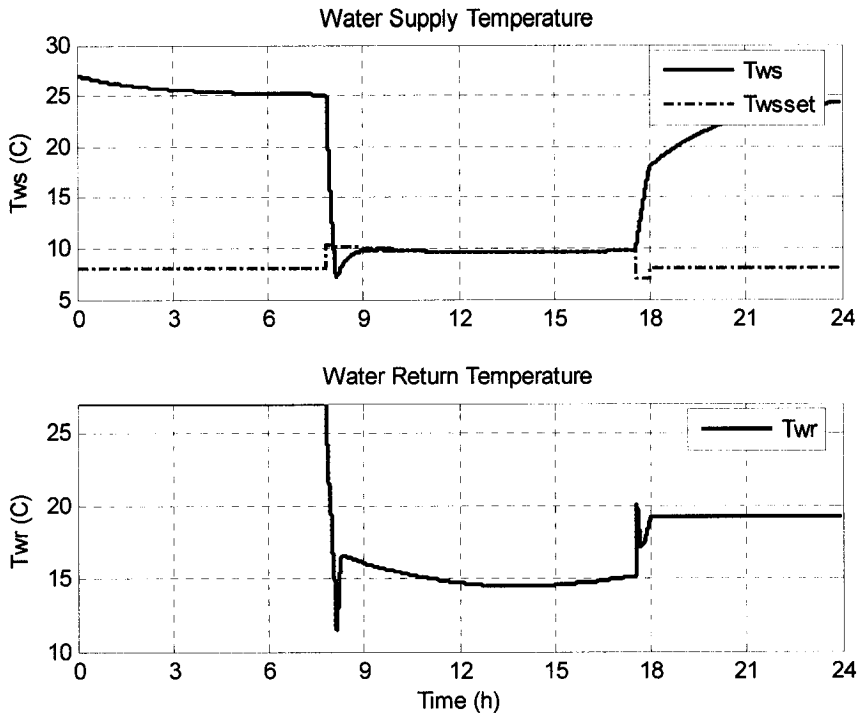


Figure 3.12. Base Case Responses for Tws and Twr

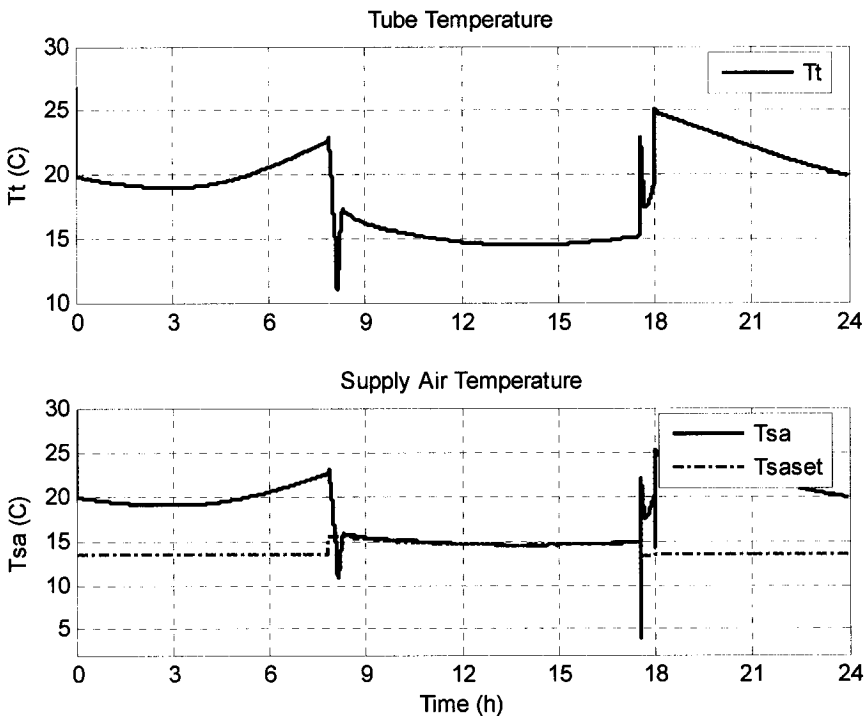


Figure 3.13. Base Case Responses for Tt and Tsa

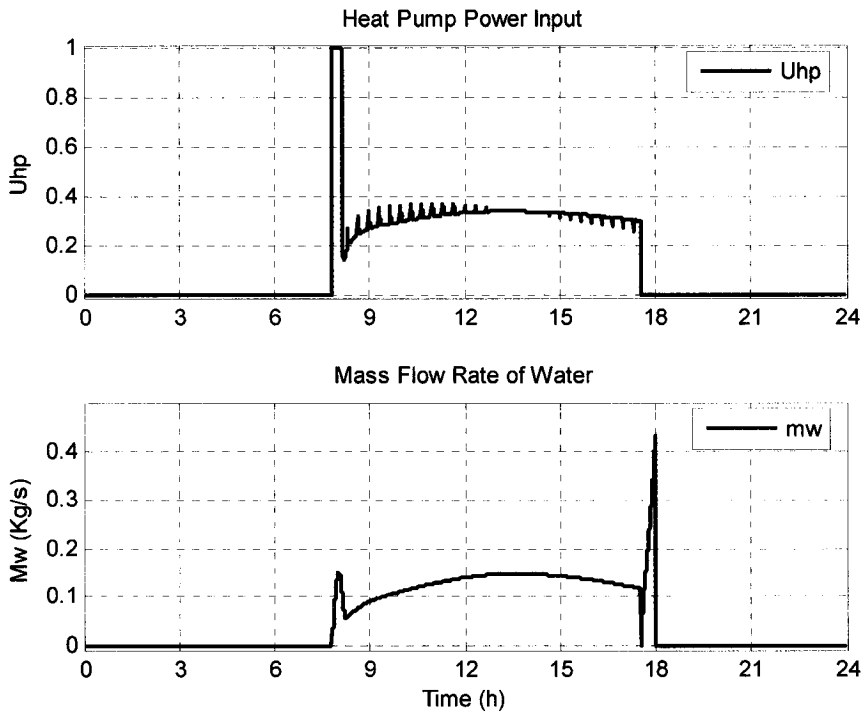


Figure 3.14. Base Case, Control Variables Uhp and mw

Figure 3.14 depicts the power input to the heat pump and the mass flow rate of water. It is easier to appreciate the changes in the set point in the supply water temperature in this figure, each time there is a change, it is translated in an increase in the power demand in the heat pump (small peaks in the power input). Also the sluggish response of the controller of the mass flow rate of water (air supply temperature control loop) is evident, since the curve is really smooth even with the ongoing changes of the set point. Since the control variable does not react fast enough, *Tsa* is unable to reach its set points. Part of the reason is the small range of the controller gains in this particular loop; nonetheless a revaluation on the set point algorithm might improve the performance of the system.

Figure 3.15 shows the percentage of outside air admitted in the building and the mass flow rate of air in the zones; since the outdoor air algorithm states that during unoccupied hours the building should profit of the lower outside temperatures, the *xv* tends to

maintain itself at its maximum: 1. The variable *Ma* is the equivalent of the sum of the individual mass flow rates of air for the individual zones, in other words the amount of air handled by the cooling coil. The output from the logic algorithm that chooses the season operational mode according to the time of the day is presented in *Figure 3.16*, giving the exact EMC – strategy that was followed during the 24h simulation run.

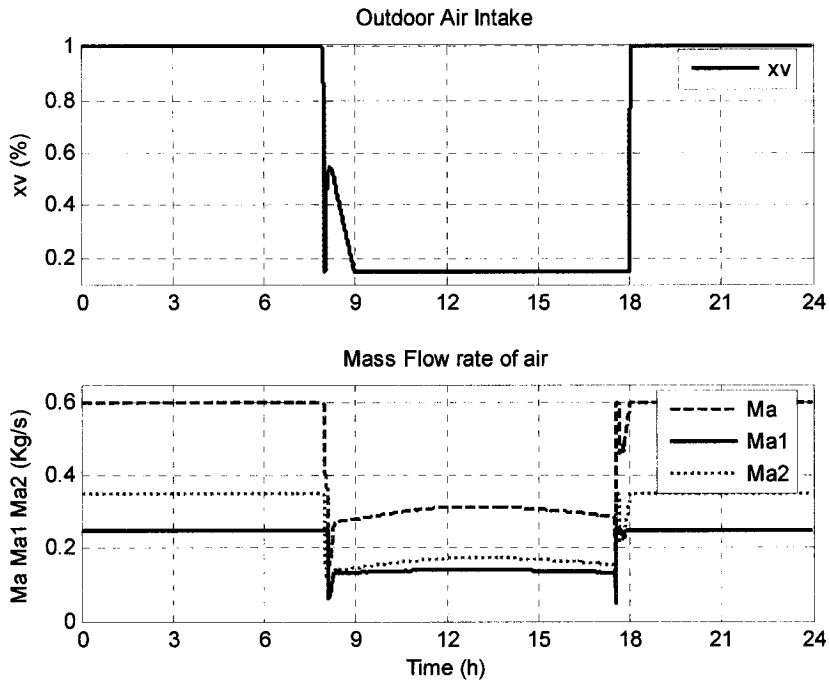


Figure 3.15. Outdoor Air Intake and Mass Flow Rate of Air in the Zones.

Figure 3.17 depicts the total energy consumption and the hourly energy consumption. This energy profiles will be used throughout this work in order to compare and quantify the potential benefits of the different strategies developed. The energy consumption of the closed loop simulation was 362.89 MJ and for the Base case was 214.05MJ, leading to savings over 40%.

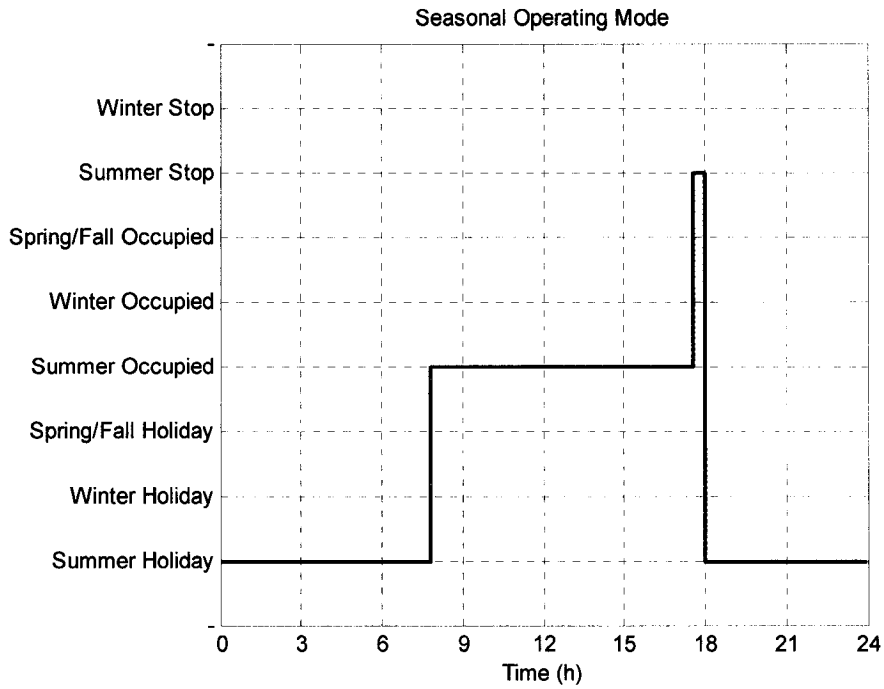


Figure 3.16. Operational Mode Season Selection

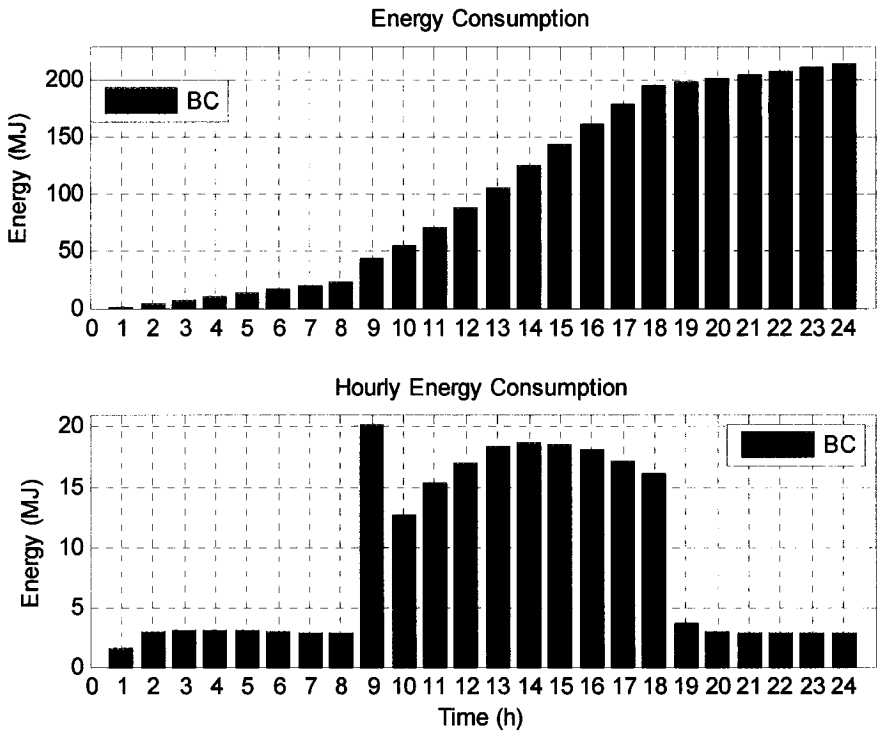


Figure 3.17. Total and Hourly Energy Consumption, Base Case

3.5 Conclusions

1) The implemented model is suitable for simulating the dynamic performance of a two zone VAV – HVAC system. However, under very low mass flow rates of water, the model yields inaccurate results like very fast changes in the variables, which are inconsistent with slow changes in real thermal systems. This could be a consequence of the relation between the mean bulk temperatures in the coil and the mass flow rate of water.
2) It was concluded that the free cooling associated with the outdoor temperature can be easily implemented with the already existent sensors used in monitoring the HVAC system of a building. The temperature based proportional band algorithm was implemented to calculate the amount of outdoor air to be admitted into the building.
3) The application of a design procedure to obtain the PI constants or the use of adaptive control methods to determine the PI controller gains is recommended to improve the overall performance of the system.
4) The energy consumption of the closed loop simulation was reduced by about 40% by implementing four EMC strategies. There is room for even further improvement not only in energy savings, but as well in the development and implementation in existing buildings.

4 ARTIFICIAL NEURAL NETWORK EMC STRATEGIES

4.1 Neural-EMC System

In this section the objective is to train and develop a set of non adaptive neural networks (NNs with fixed weights and biases) that can be implemented in an overall Neural – EMC System. To this end, first each EMC Function discussed in Chapter 3 is considered. By using the algorithms given in that chapter, simulation runs were made to generate an input – output data set. These data sets were used to train the EMC Neuro – functions.

4.1.1 Start and Stop Lead Time Neural Networks

4.1.1.1 Start Stop Lead Times Base Case and Data Acquisition

As described in section 3.4.4, the start and stop lead time algorithms are based on a reduced order dynamic model of the zone, from which a simple relationship was determined. The Lead time was found to correlate with the load in the room, the outside temperature and the building thermal mass. These equations are:

$$\begin{aligned} t_{start} &= a_1(Tz - Tzset) + a_2(Tz - Tzset)(To - Tzset) + a_3 \\ t_{stop} &= b_1(Tzfin - Tz) - b_2(Tzfin - Tz)(To - Tzset) \end{aligned} \tag{26}$$

Where *Tzfin* is the expected final temperature of the zone:

$$Tzfin = Tzset + 2 \tag{27}$$

and a_i and b_i are coefficients. These coefficients are functions of building thermal characteristics. An exact determination of the coefficients in real buildings is not an easy task. That is where the use of NNs can facilitate the implementation in real buildings through off line training or online adaptation to reduce energy consumption in existing

HVAC systems. In this section, individual non adaptive NNs were developed for each algorithm, one for start lead time and the other for the stop lead time.

In order to obtain suitable data for training the networks fifteen sets of simulations runs were made. In the simulation experiments conducted, the important operation parameters were varied between a chosen high and low limits, generating a varied input profiles while the impact in the lead time was recorded. From this data the first ten records were used to generate and train the networks and the rest were used to see the ability of the network to adapt to unseen data. The parameters to generate the input/output data set are depicted in *Table 4-1*.

Data No	*To max*	*To Min*	*Qs1 max*	*Qs1 min*	*Qs2 max*	*Qs2 min*	*Tzset1*	*Tzset2*	*Tsaset*
1	30	15	1280	500	1800	700	23	24	14
2	28	20	1280	500	1800	700	23	24	13
3	28	20	1000	500	1200	900	22	23	15
4	30	27	1150	700	1700	800	22	22	12.5
5	22	15	1150	700	1700	800	25	24	14.5
6	23	18	1223	555	1567	780	20	22	15.5
7	29	22	1223	555	1567	780	24	22	13.5
8	25	19	700	500	1000	700	23	21	12
9	25	19	700	500	1000	700	24	24	11
10	30	21	1280	800	1800	1000	22	22	12
11	27	22	1122	600	1800	1000	22	22	14
12	30	21	1200	900	1432	777	23	24	13
13	28	20	1280	800	1800	1200	23	21	13
14	27.2	19.4	1244	560	1210	880	23.5	24.2	12.8
15	28.5	18.3	1040	670	1300	990	23.3	25	14.4

Table 4-1. Input Parameter Range for Simulation Runs to Generate Data to Train the NN.

4.1.1.2 Start lead time Neural Network

For online implementation of NNs, it is important to select a simple network configuration and to choose the minimum number of inputs needed for training. To this end, a question arises as to what impact each of the operating variables shown in *Table*

4-1 above have on the prediction of lead time. To address this issue, the well known Principal Component Analysis (PCA) (Betta, G. and Pietrosanto, A, 1998) technique was applied to the data set.

The result from PCA depicted in *Table 4-2* shows that only four out of the six inputs shown in *Table 4-1* have significant impact on the start lead time. Therefore, these four inputs: outdoor temperature (*To*), supply air temperature (*Tsa*), the maximum between the two zone temperatures (*Tz*), and the corresponding expected set point for that zone once the building is occupied (*Tzset*), were used to train the Start Lead Time Neural Network (SLTNN). To keep the data size manageable, a sampling time of 30 seconds was used to carryout the principal component analysis. The minimum contribution desired was set at 3%. Accordingly the obtained results showed that the effect of zone loads *qs1* and *qs2* on the prediction of the start lead time are insignificant (less than 3%).

Variable	**Fraction Variance**
To	0.537
Tsa	0.1746
Tz	0.1501
Tzset	0.0993
qs1	0.0295
qs2	0.0095
Size Principal Component = 4	
Minimum Contribution = 3%	

Table 4-2. PCA Results for the Start Lead Time.

Consequently the input vector of the SLTNN consists of four parameters such as:

$p = [To; Tsa; Tz; Tzset]$ (30)

Where p is the input vector of the SLTNN, *To* is the outside temperature, *Tsa* is the temperature of supply air, *Tz* is the temperature of the zone, and *Tzset* is the expected set point in the zones once the building is occupied.

In the beginning the NNs were intended to be trained with the dynamic data obtained from the 24 hour simulation runs. However such an amount of information was proved to be unnecessary, when by just using the measured temperatures at the beginning (one hour before) of the start period provided good enough results. Subsequently with the simulation data acquired at the beginning of the start time, the network was able to predict adequate start lead times for the prevailing conditions in the zone. This greatly improved the chances of neural networks to be implemented in real systems, since the data required can be obtained rather easily, and although in this particular case the data to train the NNs was obtained from a simulation, the same data can be obtained from the sensors monitoring the performance of HVAC systems.

The input/output data set that was chosen to train SLTNN is depicted in *Table 4-3*:

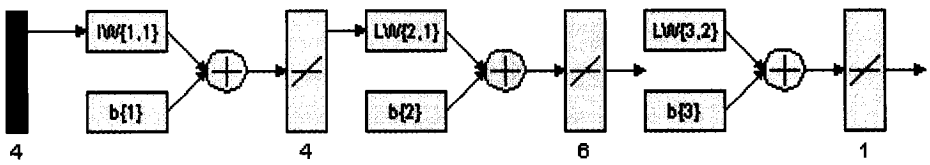
Day	***To*** (C)	***Tsa*** (C)	***Tz*** (C)	***Tzset*** (C)	**Lead Time** (s)	**Start Time** (s)	**Lead Time** (min)
1	19.9	29.8	31.6	24	1785	26985	29.75
2	22.9	36.5	38.9	24	643	25843	10.72
3	24.5	35.1	37.0	23	979	26179	16.32
4	27.9	41.4	43.8	22	1346	26546	22.43
5	16.6	30.4	32.9	24	2058	27258	34.3
6	19.2	32.2	34.5	22	1800	27000	30
7	23.9	35.9	38.1	24	1406	26606	23.43
8	21.1	31.3	33.1	21	1725	26925	28.75
9	21.6	30.5	32.1	24	1897	27097	31.62
10	23.0	39.8	42.8	22	24	25224	0.4

Table 4-3. Initial Training Data for the Start Network.

4.1.1.3 SLTNN Architecture

Several architectures were tried to determine one that will give satisfactory results. All of them were feed forward non adaptive NNs, created with the aid of the neural networks toolbox from Matlab ®. Two and three layer networks ranging between 1 to 10 neurons per layer and different transfer functions in the layers were tried. Although the difference

in the training error was not very big, the capacity of the network to react to minor changes in the inputs was the main selection criteria. The final network choice was: a feed forward, three layer network, [4, 6, 1], with 'purelin' (linear transfer function in Matlab NN tool box) as transfer function in all the layers and the Levenberg-Marquardt training algorithm, as shown in *Figure 4.1.*



Start Neural Network Architecture
No Inputs = 4
No Layers = 3
No Neurons per Layer = [4, 6, 1]
Transfer Functions in Each Layer=
{'purelin', 'purelin', 'purelin'}

Figure 4.1. SLTNN Architecture

As the name of the neural network states, the lead time is generated, this means that the neural network output will be subtracted to expected time of occupation, in this particular case 8:00 am. The results from the initial training are shown in the upper part of *Figure 4.2.* The maximum error encountered compared to the start lead time given by the base case algorithm in this initial training was of 11 minutes (643 seconds) and the average error was about 5 minutes (313 seconds). The trained network was tested with the rest of the data obtained from the simulations, to test the learning ability of the network in response to an unseen data set. The results of the predictions for the SLTNN are compared with the data from the base case in the lower part of *Figure 4.2.*

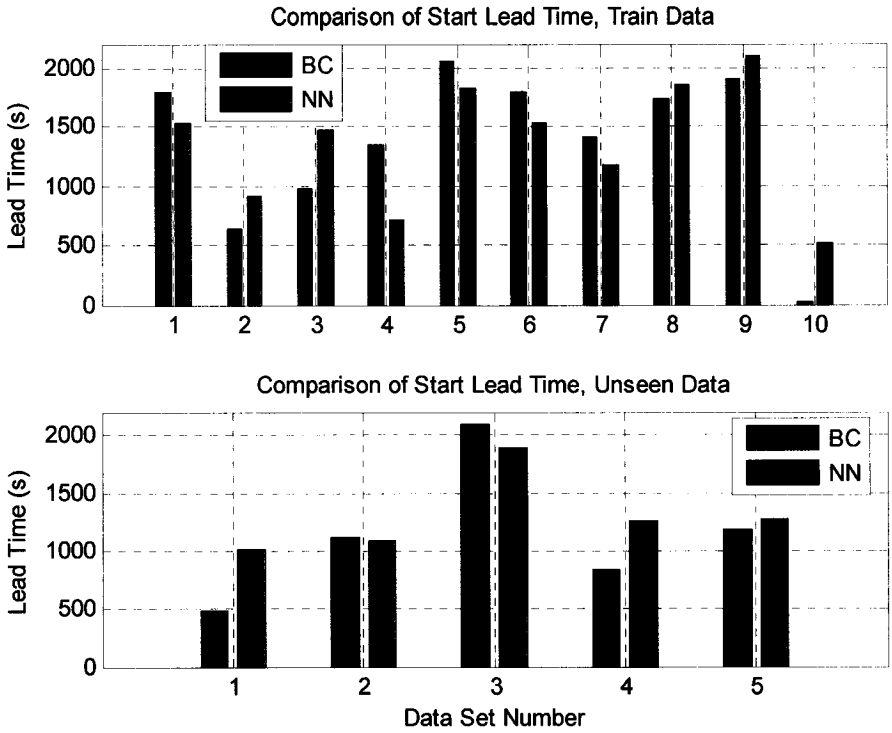


Figure 4.2. Results for the Training SLTNN

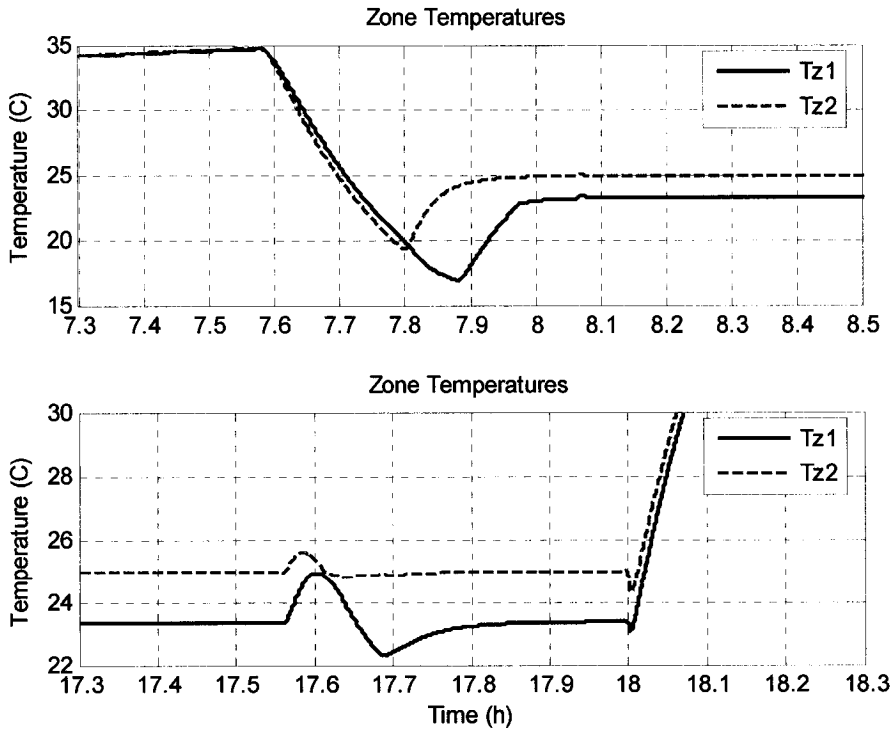


Figure 4.3. Results From the SLTNN and STPNN with Unseen Data

The maximum error for this trial run was of 9 minutes (544 seconds) and the average error was 4.5 minutes (269 seconds). *Figure 4.3* shows the results obtained with the Start and Stop lead time NNs integrated in the simulation. From the figure is clear that the SLTNN properly selected the lead time in order to have the zone with the highest temperature at the start time (7am), in this case *Tz2,* in its corresponding set point at the beginning of the occupied period (8am).

It is worth to take into account, that since we use the same plant for two zones, one of the two zones is most likely not going to achieve the set point in the moment that the building is occupied. This is due the way that the algorithms for the base case were selected (See Chapter 3). For control purposes, just one of the two zones is selected to determine the heat pump input; in the winter it is the zone with the lowest temperature, which will require the major energy input to reach its set point, and during the summer the zone with the highest temperature, will be the one selected for the management of the *Tws*. This will lead to minor discomfort to the occupants of the other zones. The time difference for the second zone to achieve its set point is not big, and the temperature difference is usually less than an acceptable 1.5 C from the desired set point in the zone.

4.1.1.4 Stop Lead Time Neural Network

For the Stop Lead Time Neural Network (STPNN) a similar analysis was used. The result from the principal component analysis depicted in *Table 4-4* showed that zone loads *qs1* and *qs2* do not influence the prediction of stop lead time.

As such the four inputs to the STPNN selected were:

p = [*To; Tsa; Tz; Tzset*] (31)

Variable	Fraction Variance
To	0.4133
Tsa	0.3338
Tz	0.1692
Tzset	0.0601
qs1	0.0191
qs2	0.0045
Size Principal Component = 4	
Minimum Contribution = 3%	

Table 4-4. PCA Results for the Stop Lead Time

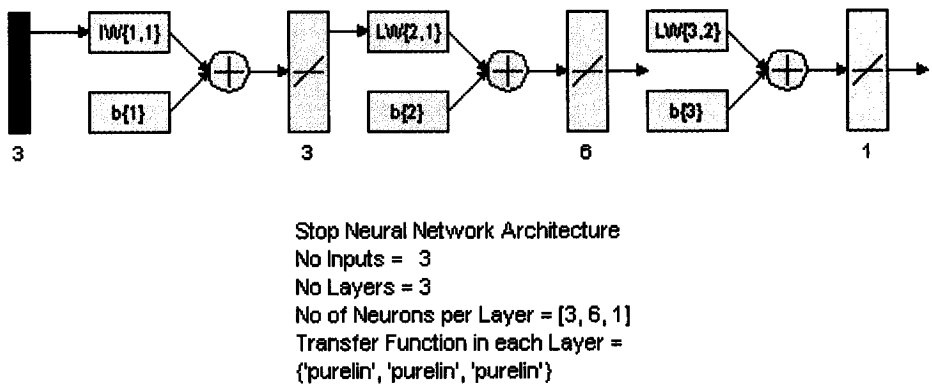


Figure 4.4. Stop Lead Time Neural Network Architecture

However, when trying different architectures for the stop lead time neural network, the best results were obtained with a 3 layer network, [3, 6, 1], all transfer functions were 'purelin' , with the Levenberg-Marquardt training algorithm , but only with three inputs:

$$p = [To;\ Tsa;\ Tz] \tag{32}$$

The reason for this is fairly simple. The PCA was used with the dynamic data, but as we explained in the later section, for training purposes it was more effective just to use the initial condition when the algorithm is about to start. Since the zone is in a steady state

due to the controllers for the zone temperatures, the value of *Tzset* was actually very close to *Tz*, subtracting the contribution for the NN from that variable. The initial data set used for the training and creation of the network is depicted in *Table 4-5*.

Day	***To*** (C)	***Tsa*** (C)	***Tz*** (C)	**Stop Time** (s)	**Lead Time** (s)	**Lead Time** (min)
1	27.2	12.9	24	63227	2027	33.8
2	26.5	12.9	24	63205	2005	33.4
3	25.6	14.9	23	62327	1127	18.8
4	29.7	12.4	22	62063	863	14.4
5	21.6	14.5	25	62100	900	15
6	21.9	15.5	22	61800	600	10
7	28.1	13.4	24	63172	1972	32.9
8	24.0	12.0	23	63581	2381	39.7
9	24.2	11.0	24	63062	1862	31.0
10	28.4	11.9	22	63500	2300	38.3

Table 4-5. Initial Training Data for the Stop Network.

In this case the stop lead time is calculated, again adding the initial stop time (61200 seconds) to the results obtained from the STPNN. *Figure 4.5* displays the results of the supervised training, obtained by simulating the data in *Table 4-5*. Once more the maximum error encountered is about 15.2 minutes (912 seconds) and the average error is 9.5 minutes (570 seconds). The STPNN was also tested using the last five (unseen) data sets. We can see that the maximum error decreased to 13.5 minutes (814 seconds) and that the average error is also lower: 7 minutes (421 seconds). Although the maximum and average error decreased during the testing, is expected to have high errors while comparing with the base case algorithm. However, when using the SLTNN and STPNN integrated with the simulation the zone temperatures are kept in the ± 2C target range from their respective set points. The lower part of *Figure 4.3* shows the simulation run of

the STPNN integrates in the simulation. There is noticeable change in the zone temperature when the chiller is turned off at the selected stop lead time given by the NN.

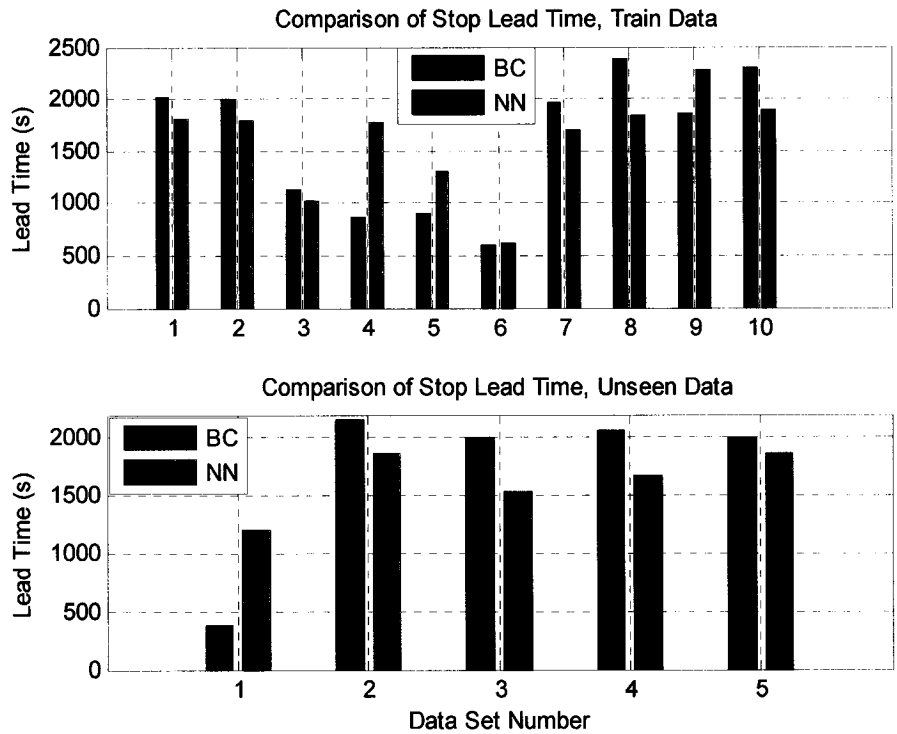


Figure 4.5. Results for the Training of STPNN

As explained in Section 3.4.5, during the rest of the stop time, the pump is left on to use the energy stored in the water tank. This gives the system the possibility to the zones to maintain close to the selected set points until the building is unoccupied, while reducing the energy consumption in the system originated in the heat pump. Furthermore, as a precaution an upper limit on the stop lead time was imposed using a saturation algorithm. This will be helpful in real buildings to avoid accidental shutdown of the HVAC system based in unreasonably high stop lead times.

4.1.1.5 Discussion

The main objective of developing Neural – EMC functions is to be able to implement them in real HVAC systems in buildings. Lacking this, on alternative method of testing the performance of Neural – EMC networks was used. A dynamic model of two – zone variable air volume HVAC system with four local control loops was developed. This model was used to simulate 24 – hour operation of the building HVAC system.

To this end, the dynamic model acted like a virtual building – HVAC system. The developed SLTNN and STPNN were integrated in this model and typical 24 – hour operation was simulated. The results show that the integrated simulation model with embedded NNs is performing satisfactorily in predicting the systems start and stop times. The simulation results depicted in *Figure 4.3*, correspond to a warm day with maximum outdoor air temperature of 29C and a minimum of 18C. The predicted start and stop times from the respective neural networks, were 7:32 am (27105s) and 5:26pm (62735s) respectively.

In real systems it is time consuming to determine the building coefficients that will establish relationships for the building's thermal behaviour. Furthermore, these coefficients are dependent on the weather conditions and heat transfer coefficients inside the zones that can and will change with the time, and thus influencing the prediction of the base case algorithms (See Equations 26 and 27). On the other hand is possible to use the knowledge of the building operators and some measurements during the normal function of the system in order to produce the initial data needed to train these NNs. The proposed NNs require just the measurements of outside temperature, the zone temperature, the desired set point and the supply air temperature. This would make the

instrumentation of the energy management strategies, more accommodating for different buildings. By determining the time that the HVAC system uses to reach the set points (for the particular case of these simulations one hour was used), and monitoring the data for a relatively short period of time (10 to 15 days) it may be possible to develop a rather stable energy management strategy that will provide start and stop lead times with good results.

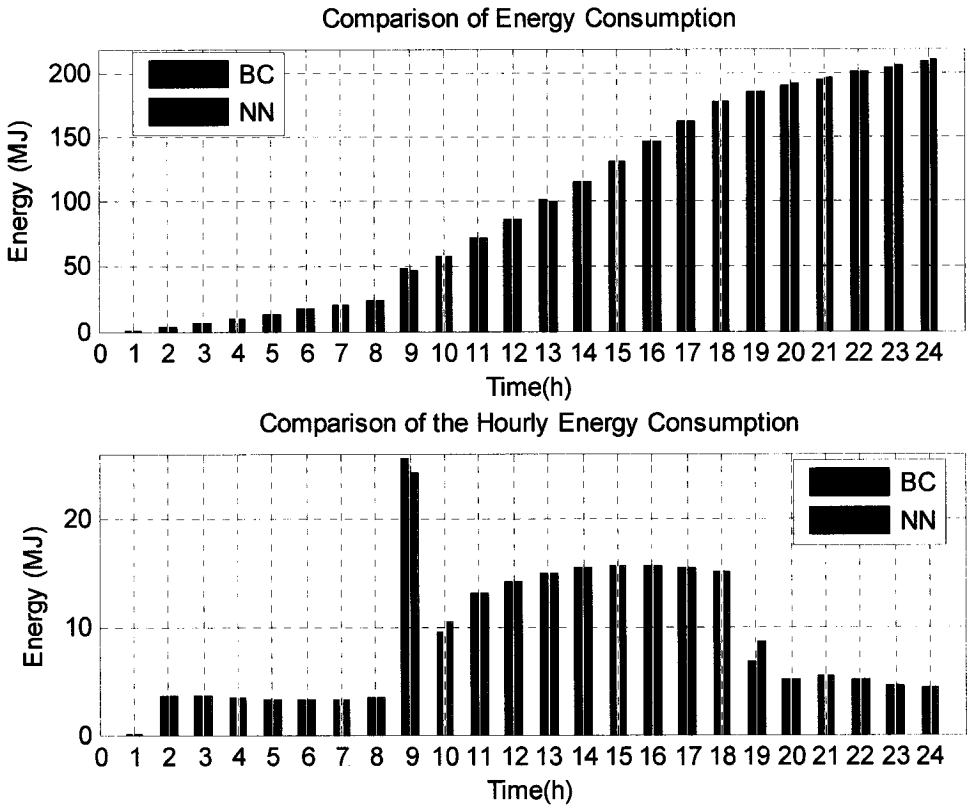


Figure 4.6. Comparison of the Total Hourly Energy Consumption of the NN with the Base Case

Figure 4.6 shows the comparison of the energy consumption of a typical 24 – hour simulation run, between the base case and the embedded SLTNN and the STPNN. As expected the main differences in the energy consumption are after the start and stop times. Although these differences seem apparent, comparing the total energy consumption, as

shown in the lower part of the same figure, the total difference is almost negligible (1.3 MJ less consumed by the simulation with the neural networks). However the relative easiness in the implementation, still gives a potential advantage to the developed non adaptive NNs. Another avenue to pursue is to develop these energy management strategies using adaptive NNs. In the following chapter the application of adaptive neural networks in EMC systems will be explored.

4.1.2 Outdoor Air Economy Cycle Neural Network

4.1.2.1 Outdoor Air Economy Cycle (Base Case)

The outdoor air economy cycle as explained in Chapter 3, is temperature based algorithm. The objective is to maximize free cooling whenever the outdoor air temperature is close to the supply air temperature. This particular algorithm is a vital component for the energy savings during summer, spring and fall operations. During the summer, the algorithm allows the system to use the lower outside air temperatures during the night to maintain the building at reasonable temperatures with the heat pump turned off. This "free cooling" is also used in the Spring and Autumn, when the outdoor temperatures can be the same, or even lower as the required supply air temperature.

Equation 22 shows the proportional band control algorithm that is the one used in the base case.

$$x_v = \begin{cases} Tz - n < To < Tz + n; & x_v = x_{v\min} + (1 - x_{v\min})\left(\dfrac{Tz - To}{n}\right) \\ To \le Tz - n; & x_v = 1 \\ To \ge Tz + n; & x_v = x_{v\min} \end{cases} \qquad (22)$$

$$n = 2$$

Where n is the proportional band, *To* is the outdoor air temperature, *Tz* is the temperature of the zone, considered the same return air temperature (no temperature losses in the ducts are considered), x_{vmin} is the minimum amount of outdoor air that should admitted in the building (MNECCB, 1997), and x_v is the actual percentage of outdoor air admitted in the system.

To create the neural network for this EMC – function, it is necessary to have a suitable set of data. Since this algorithm works 24 – hours, with just one day simulation run we acquire enough data to train the neural network. The main conditions for the data sets are shown in *Table 4-6.*

Data No	To max	To min	Qs1 max	Qs1 min	Qs2 max	Qs2 Min	Tzset1 max	Tzset1 min	Tzset2 max	Tzset2 min	Tsaset max	Tsaset min
1	30	15	1280	500	1800	700	25	18	25	18	14.5	10
2	28	20	1280	500	1800	700	25	18	25	18	14.5	10

Table 4-6. Principal Data Used to Create the Training Data Set for the Outdoor Air Economy Cycle Neural Network.

To include changes in the temperatures of the zones and the temperature of supply air, these variables' set points were changed during the 24 – hour simulation run. This is to avoid the neural –EMC to follow a certain pre - established pattern, that would limit the generalization capabilities of the developed non adaptive NN. Also a different set of conditions was generated to test the learning ability of the NN.

From all the data seven variables were chosen according to the possible influence on the outdoor air economy cycle: the outside temperature *To*, the zone temperature *Tz*, the set point of the zone *Tzset*, the supply air temperature *Tsa* and it's set point *Tsaset*, plus the cooling load in the two zones *qs1* and *qs2*.

Figure 4.7 depicts the outside air temperature *To*, along with the zone temperatures, the supply air temperature and the corresponding percentage of outdoor air admitted in the building (*xv*).

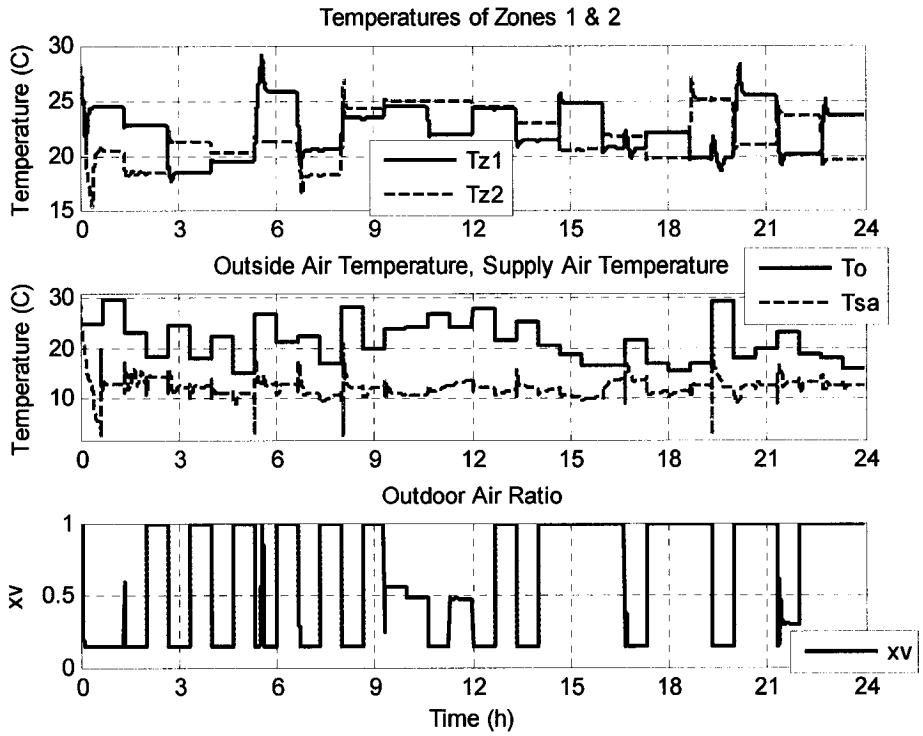


Figure 4.7. Data generated for the outdoor air economy cycle: xv

Variable	Fraction Variance
To	0.4853
Tz	0.2562
Tsa	0.1342
Tzset	0.0773
Tsaset	0.0470
qs1	0.0000
qs2	0.0000
Size Principal Component = 3	
Minimum Contribution = 10%	

Table 4-7. PCA Results for the Outdoor Air Economy Cycle

The principal component analysis (PCA) was used for this data, to provide an appropriate input vector for the outdoor air economy cycle neural network (xvnn) according to the influence of each variable. A minimum of 10% influence was considered to include a parameter in the input vector. *Table 4-7* shows the results from the PCA for the selected variables. With these results the first three of the seven variables were selected(*Table 4-7*). The set points of the zone temperatures and the supply air temperature were not included in the input vector for the network for two reasons: In the data set selected the set points and the corresponding temperatures are almost the same through the selected simulation, so its contribution is minimal. Also is worth to note that during the combined operation of the EMC – functions this algorithm will work independently from the set point, this is why it would be misleading information for the neural network when the combination of the Neural – EMC – functions is completed. Then the input vector that is selected for *xvnn* is shown in Equation (33).

$$p = [To\ ;\ Tz;\ Tsa] \tag{33}$$

Where *p* is the input of the neural network, *To* is the outside temperature, *Tz* is the maximum zone temperature, and *Tsa* is the supply air temperature.

4.1.2.2 Outdoor Air Economy Neural Network: *xvnn*

After deciding the size of the input vector from the results of the principal component analysis, a series of different architectures were tried, varying the number of layers, the number of neurons per layer and the transfer functions used, starting from the simple architectures to more complex ones. Taking in account the adaptability as well as the performance of the neural network, a two layer [3, 1] feed forward neural network, with

the Levenberg-Marquardt training algorithm was selected. As transfer function for both layers 'tansig' the hyperbolic tangent sigmoid transfer function of the Matlab neural network toolbox was used. This function calculates its output according to Equation (34).

$$a = \frac{2}{\left(1 + \exp^{(-2*n)}\right)} - 1 \tag{34}$$

Where a is the output and n is the input of the transfer function (Pingkang, L. *et al.*, 2002). *Figure 4.8* depicts the general architecture of the outdoor air economy cycle neural network (*xvnn*).

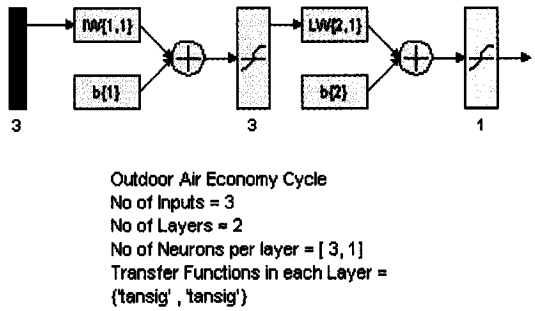


Figure 4.8. Outdoor Air Economy Cycle Neural Network Architecture, xvnn

The training data set was a vector of 156,393 samples for each one of the selected input variables. The *xvnn* surpasses the chosen training goal of 0.001 in fourteen epochs, which is a reasonably fast training time. The error comparison between the data generated by the base case and the NN was calculated by simply subtracting these results (base case – NN). The maximum error found was 0.1379, and the average error was 0.0119.

Using a similar profile for the outdoor temperature and zone temperatures another data set was generated to test the generalization capabilities of the NN. The error related to the performance of the NN was very close to that encounter with the training data set, with a maximum of 0.1197 and an average of 0.0121. While the maximum error was lower than

the one in the original training the average error increased. This can be attributed to the frequent changes in the set points of the variables included in the input vector for the *xvnn*. Furthermore, the transfer function selected for the neural network, 'tansig' (Hyperbolic tangent sigmoid transfer function in the Matlab neural network toolbox) is nonlinear, while part of the algorithm used for the base case is. This is clearly seen in the error depicted in *Figure 4.9*, where the biggest differences are in the periods of time where the zone temperature is close to the outdoor air temperature. During this time the proportional band algorithm uses a linear relationship to determine the amount of outdoor air admitted in the building. *Figure 4.9* shows the comparison of the neural network results with the results obtained from the Base Case algorithm, in a typical 24-hour simulation run where only the EMC-function of outdoor air economy cycle is used, also the error is depicted in the same graph. This second set of unseen data is generated with a normal 24 – hour simulation run and, with the outdoor air temperature profiles from chapter three and with only the outdoor economy cycle from the EMS. The maximum error for this data was 0.2081 and the average error was 0.0108. Having a constant set point during the 24 – hour simulation for these variables, the network only has to deal with the changes associated to the outside temperature. *Figure 4.10* shows the difference between the energy consumption of the base case proportional band algorithm and the energy consumption with the Neural – EMC function. This difference in energy consumption is negligible.

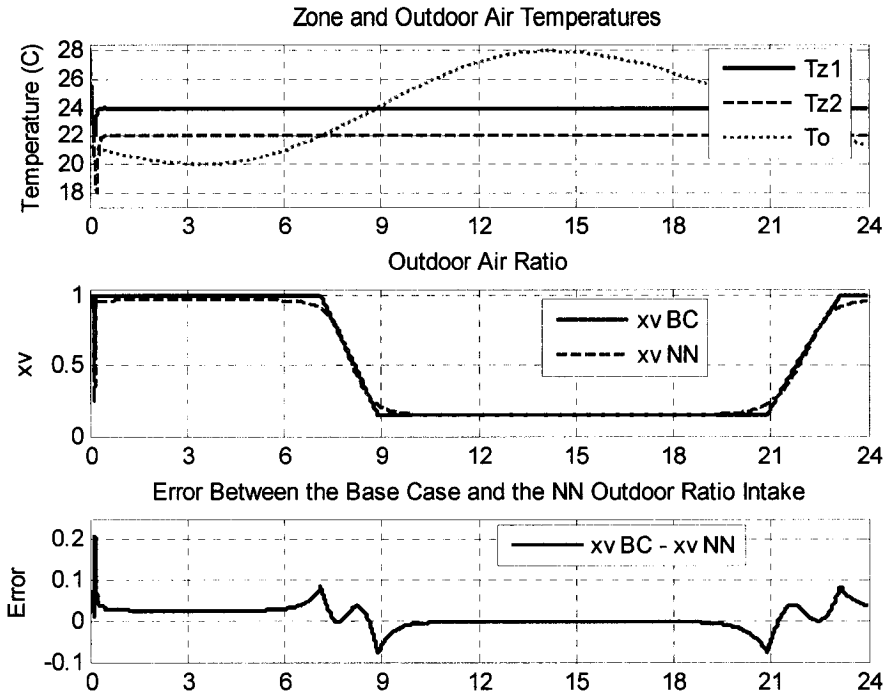


Figure 4.9. Comparison of the XvBC and by xvnn with an Unseen Data Set

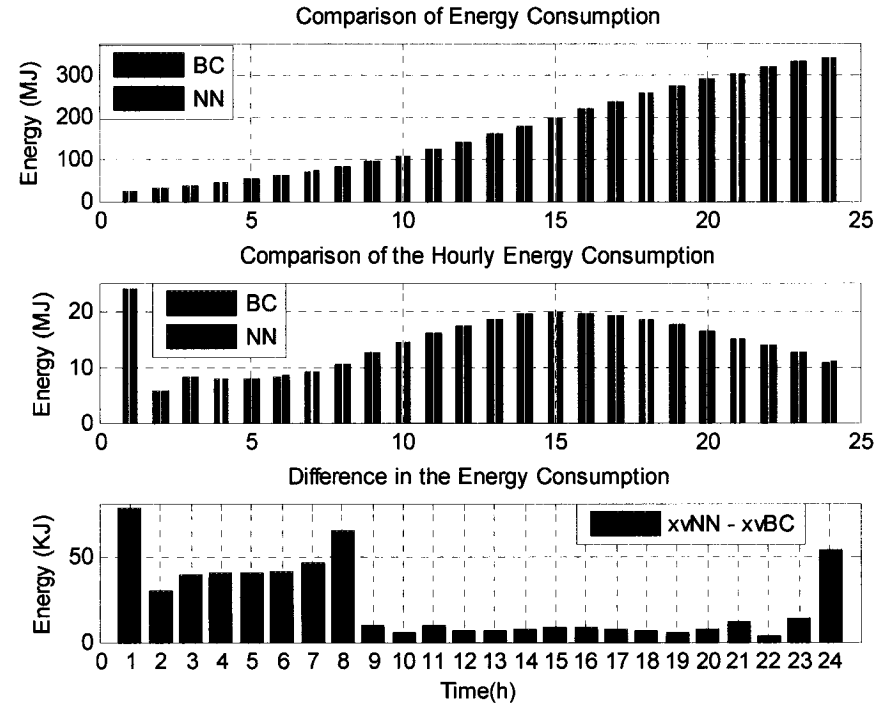


Figure 4.10. Comparison of the Energy Consumption of XvBC and by xvnn

4.1.3 Temperature Reset Strategy

4.1.3.1 Reset Strategies for Supply Air Temperature and Water Supply Temperature

This algorithm changes the Supply air temperature set point, to reduce the energy consumption. As explained in Chapter 3, three methods were evaluated. Equation 23 describes the method presented by Huang (2003).

$$T_{saset} = \begin{cases} T_{z,l} < T_z < T_{z,h} & T_{saset} = T_{zset} - \left(\frac{q_{s\max}}{M_{a\max} c_{pa}} \right) \left(a_1 T_o^2 + a_2 T_o + a_3 + a_4 \left(T_{zset} - T_{o\_mean} \right) \right) \\ Tz \le T_{z,l}; & T_{saset} = T_{saset,\max} \\ Tz \ge T_{z,h}; & T_{saset} = T_{saset\min} \end{cases} \tag{23}$$

Where T_z, $T_{z,l}$, $T_{z,h}$, are the actual temperature, the lowest temperature and the highest temperature of the zone respectively while T_{zset} is the maximum set point in the zones. T_{saset}, $T_{saset,max}$, $T_{saset,min}$ are the actual, the maximum and the minimum supply air temperature set points. q_{smax}, is the maximum sensible cooling load, M_{amax} is the maximum mass flow rate of air in the zones, and c_{pa} is the specific heat of the air. The equation presented is a function of the cooling load in the building and a series of constants that predict the outdoor air temperature. Since the data generated by this method would show a high dependence on the cooling load in the zones and in the prediction of the outdoor air temperature, it is most likely to obtain a NN also dependent on the same variables if we use this data set to train it. For this reason, different simulation runs were made with the objective of fitting data generating a polynomial relationship to obtain the set points dependent on other variables.

Equations 24a and 24b show the results of fitting the data for the outdoor air temperature (Appendix 2).

$$Tsaset = -0.2 \cdot To + 17.8746 \quad (24a)$$
$$Twsset = -0.18 \cdot To + 13.6487 \quad (24b)$$

The Base Case used in this study uses Equations 24a and 24b for determining the changes in the supply air temperature and water supply temperature set points. To avoid dependencies on the outdoor air temperature profile of the training data set, simulation runs were made changing the cooling loads and the outdoor air temperature every hour and a half. *Figure 4.11* shows the generated training data set for the both the supply air temperature reset NN (*tsann*) and the water supply temperature reset NN (*twsnn*).

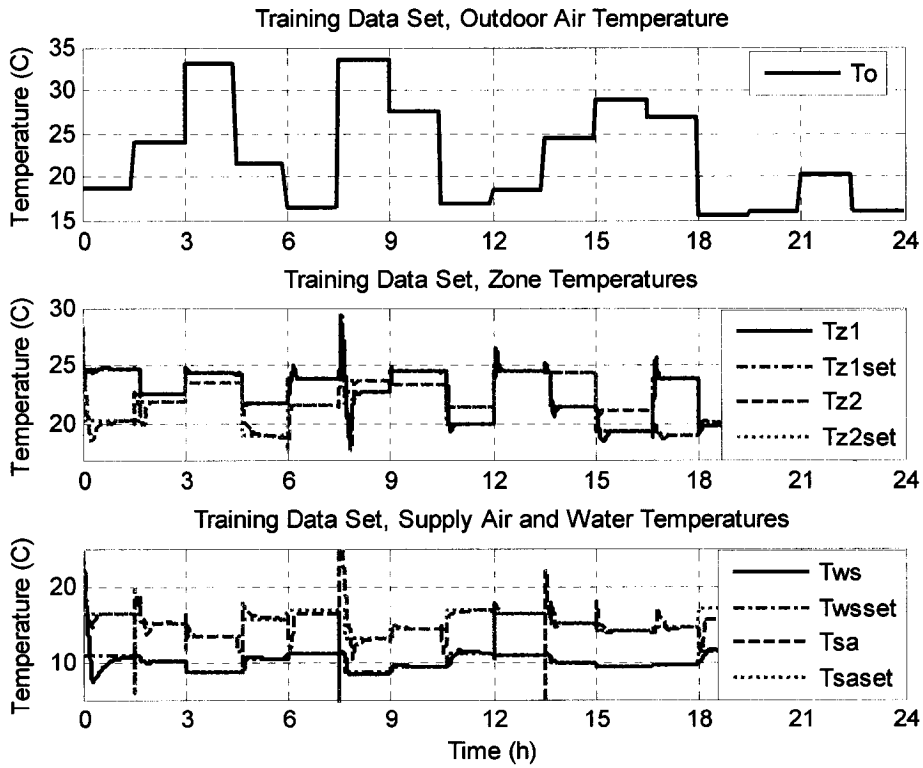


Figure 4.11. Training Data Set for the Supply Air and Water Temperature Reset NN.

With the generated data set Principal Component Analysis (PCA) was used to obtain an adequate input vector for each of the NNs.

PCA for *tsann*		PCA for *twsnn*	
Variable	**Fraction Variance**	**Variable**	**Fraction Variance**
To	0.3594	*To*	0.3402
Tws	0.2265	*Tz*	0.2225
Tz	0.1561	*Tsa*	0.1696
mw	0.1366	*Mw*	0.1523
qs1	0.0621	*qs1*	0.0651
Tzset	0.0520	*Tzset*	0.0427
qs2	0.0073	*qs2*	0.0076
Size Principal Component = 4		Size Principal Component = 4	
Minimum Contribution = 10%		Minimum Contribution = 10%	

Table 4-8. PCA Results for the Reset Supply Air Temperature and for the Reset Supply Water Temperature

The selected variables for the *tsann* PCA shown in *Table 4-8* were: the outside air temperature (*To*), the cooling loads (*qs1, qs2*) the maximum zone temperature (*Tz*) and its corresponding set point (*Tzset*), the mass flow rate of water (*mw*). For the Supply air temperature set point the water supply temperature (*Tws*) was used, while for the water supply temperature set point, also shown in *Table 4-8*, the supply air temperature (*Tsa*) was employed. A minimum contribution of 10% was set from the selected variables to be part of the input for both NNs. The resultant input vectors for the *tsann* and for the *twsnn* are shown in *Table 4-9*.

Neural Network	Input Vector	
tsann	p = [*To; Tws; Tz; mw*]	(35a)
twsnn	p = [*To; Tsa; Tz; mw*]	(35b)

Table 4-9. Input Vectors for tsann and twsnn from the PCA

4.1.3.2 Reset Supply Air Temperature Neural Network: *tsann*

With the input vector for the NN determined form the PCA and the generated data set, several architectures were tried to determine one that will give logical results, without over fitting the data set. Varying the number of layers, the number of neurons per layer and the transfer functions used, starting from the simple architectures to more complex ones, the final architecture selected for the Supply air temperature set point neural network was a three input, p = [*To; Tws; Tz*], three layer [3,6,1] feed forward neural network, with the Levenberg-Marquardt training algorithm. As transfer function for all the layers 'purelin' the linear transfer function of the Matlab neural network toolbox was used.

Figure 4.12 depicts the general layout of this neural network. It is worth to note that the final architecture uses only three of the four recommended outputs from the PCA. While testing different architectures it was found that the selected one gave close results than those generated by the four input, p = [*To; Tws; Tz; mw*], three layer, [4,6,1], feed forward neural network. Having more inputs the NN that includes *mw* requires more epochs to train, while the difference in the error achieved by both neural networks is negligible.

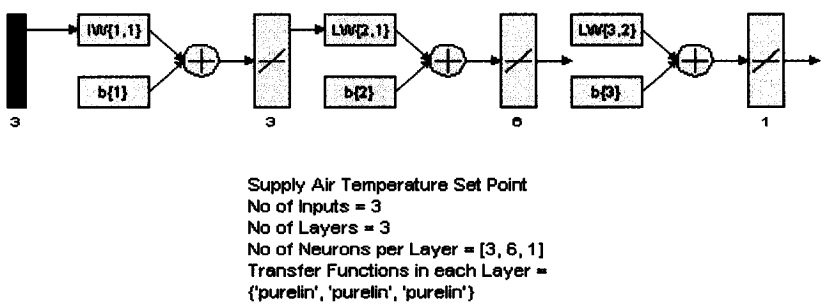


Figure 4.12. Architecture for tsann.

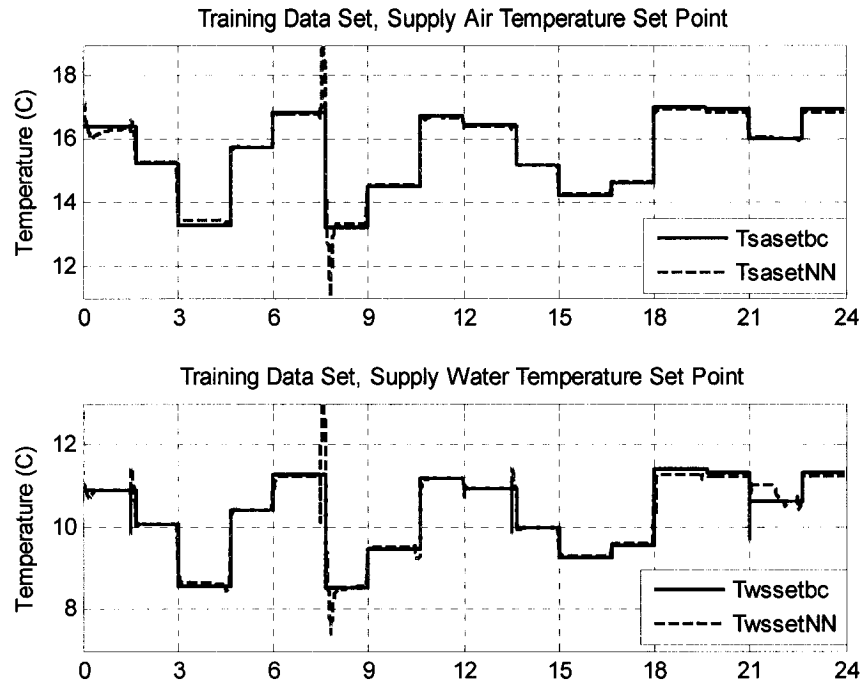


Figure 4.13. Comparison of the Set Points From tsann and twsnn with the Base Case.

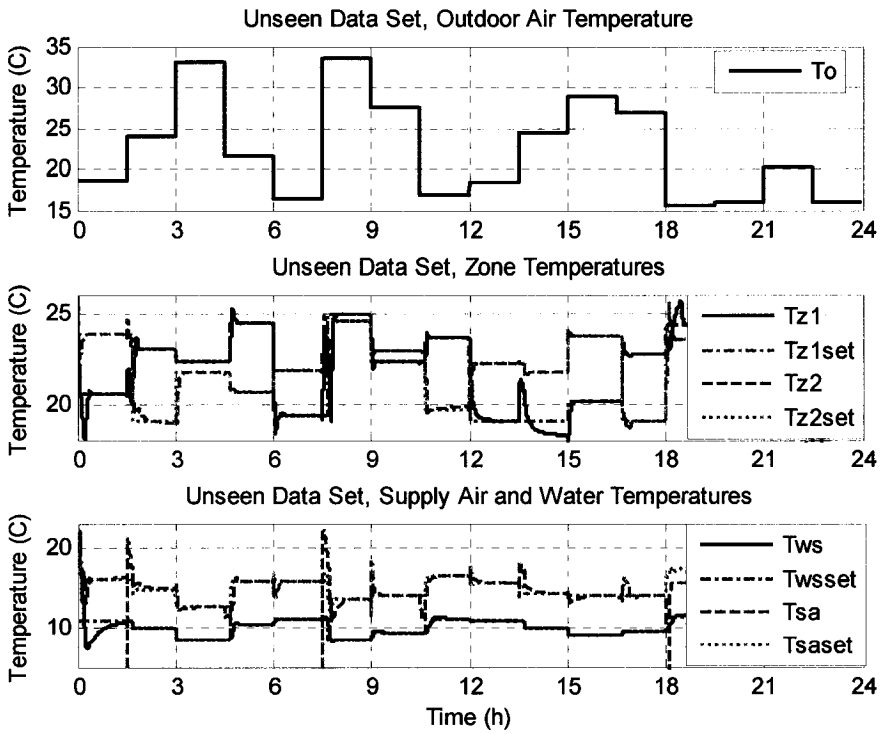


Figure 4.14.Results Obtained by the tsann and twsnn with Unseen Data.

Due to the size of the generated training data (166,030 points) a sample time of 30 seconds was chosen to reduce the size of the input vector (5776 points). The results from the supervised training are depicted in *Figure 4.13*. The maximum error between the results obtained by the Base Case algorithm and the *tsann* was 3.14 C while the average error was 0.15C. This error is expected to be reduced once the simulation is run under normal conditions, since the changes in cooling loads and outdoor air temperature will be gradual. The results obtained by the *tsann* with a set of unseen data are depicted in *Figure 4.14*. The maximum error between the Supply air set point calculated by the NN and by the Base Case algorithm was 3.9C while the average error was 0.5C. It is important to note that in most of the cases the set point of the zones was achieved and that the comfort of the occupants was not compromised by the changes in the supply air temperature.

4.1.3.3 Reset Supply Water Temperature Neural Network: TwssetNN

After the comparison of the results from several architectures, the input vector for the NN determined from the PCA proved to be correct, but unnecessarily large. The results for the final architecture selected for the Supply water temperature set point neural network (*twsnn*) was a three input, p = [*To; Tsa; Tz*], three layer [3,6,1] feed forward NN, with the Levenberg-Marquardt training algorithm. As transfer function for all the layers 'purelin' the linear transfer function of the Matlab neural network toolbox was used. *Figure 4.15* depicts the general layout of this neural network.

The final architecture uses only three of the four recommended outputs from the PCA. As for the *tsann* the NN architectures that were tried with the four inputs, p = [*To; Tsa; Tz;*

mw], gave good results, but the computational cost of processing an extra input, is not justifiable by the improvement in the performance of the NN.

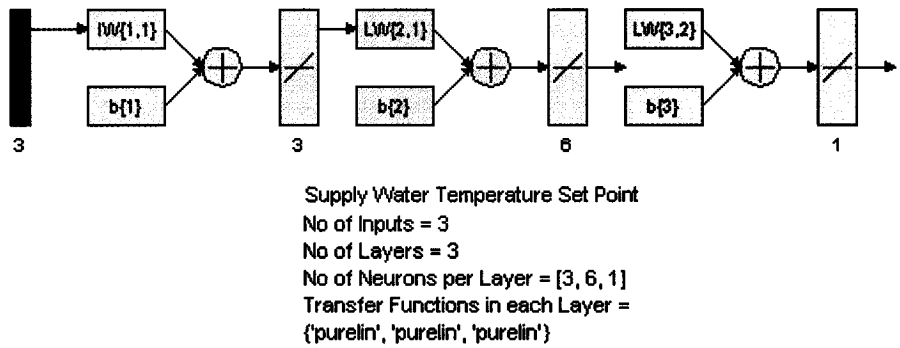


Figure 4.15. Architecture for twsnn.

After the creation of the *twsnn*, the results are compared with those obtained by the base case algorithm. This comparison is depicted above in *Figure 4.13*. The maximum error between the results obtained from the base case algorithm and the *twsnn* was 3.67C and the average error was 0.12C. As for the *tsann* an improvement in the performance of the NN is expected once the changes in the outdoor air temperature are not so brusque. *Figure 4.14* depicts the results obtained with the *twsnn* with a set of unseen data. The maximum error between the results obtained from the base case algorithm and the NN for the unseen data set was 3.4C and the average error was 0.75C.

4.1.3.4 Discussion

To compare the results from the NNs with the base case algorithm, a 24h - simulation run was made using typical outdoor air temperature and the cooling load profiles. The results obtained for both neural networks for the zone Temperatures are depicted in *Figure 4.16*. As it is expected the neural networks provide adequate set points in order to maintain the desired zone temperatures. The maximum error between *tsann* and the base case was

2.65C and the average error was 0.083C. The maximum error encountered for *twsnn* was 1.55C and the average error was 0.02C. As expected the error for the generated set points diminished with the use of a profile with the outdoor air temperature instead of the random values. Furthermore the error can also be attributed to the slower change in the set points for the base case algorithm. This is due the sole dependence of the polynomial equations on this variable, which is only considered every 1200 seconds, while the NNs are continuously changing to accommodate the change in the system. *Figure 4.17* depicts the energy consumption difference between the base case algorithm and the integrated *tsann* and *twsnn*. From the figure it is clear that the difference is not significant, the maximum error encountered was -0.1821 MJ and the average error was -0.0112MJ. The ability of the network to accommodate changes opens the possibility of its use in real systems with similar results as the base case algorithm. The question is how to implement the NNs in a real HVAC system, for the selection of the adequate set points for the supply air temperature and the supply water temperature. Since there is no need to measure or to know the cooling load in the system, the knowledge of the operators of the system and the information collected from the usual sensors to control and supervise the HVAC system would allow the NN to be trained and possibly yield good results. However the static nature of the weights of these particular NNs gives an opening to explore the possibility of a set of adaptive NNs, which would learn and accommodate to the needs of any new conditions or changes in the HVAC system.

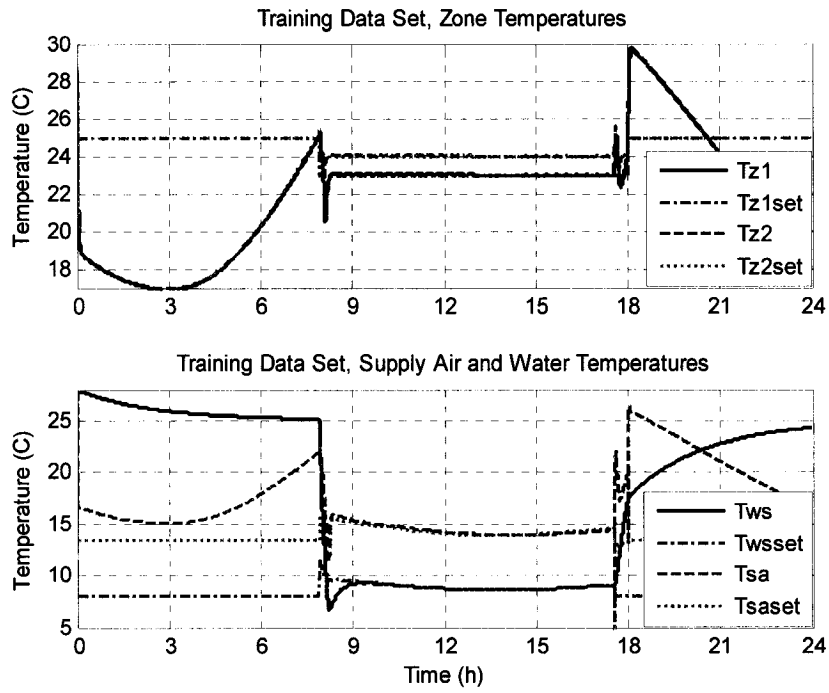


Figure 4.16.Results of tsann and twsnn with Unseen Data.

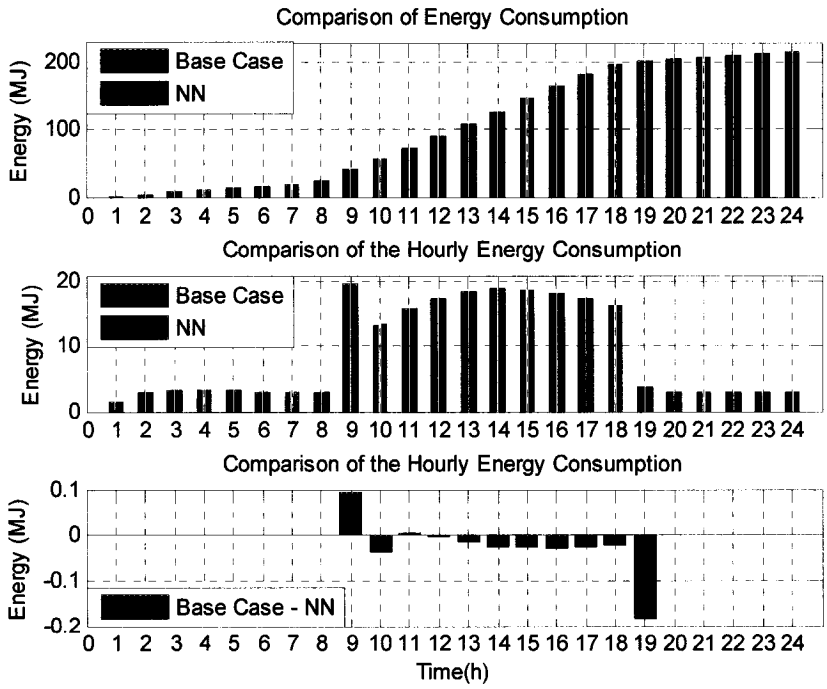


Figure 4.17.Energy Consumption Difference Between the Integrated tsann/twsnn and the Base Case Algorithm.

4.2 PI-Neuro-controllers

4.2.1 Introduction

As presented in Section 3.3.2, the system has four main control loops (*Table 3-9*). The nonlinear nature of the HVAC systems always leads to problems when selecting the gain constants for the PI controllers that regulate the system. It was shown in Chapter three that a better method to select the gain values is required in order to improve the overall performance. Since the design procedure to obtain these constants is dependent on the initial conditions selected and is highly sensitive to any changes in the system dynamics, the logical choice is to have the gain values change with time to accommodate the different operating condition and requirements of the building. With this in mind a series of NNs were developed in this chapter for the four main control loops in the system to select and change with time the gain values of the corresponding PI controllers: Supply Water Temperature Control Loop (*uhpnn*), Supply Air Temperature Control Loop (*mwnn*), Zone 1 Temperature Control Loop (*ma1nn*) and Zone 2 Temperature Control Loop (*ma2nn*). These loops were separated to create the different NNs and after corroborating the results obtained, the NNs were combined in one single model that is presented in the last part of this section.

4.2.2 Constant Gain PI Controllers (Base Case)

To initiate the process of developing of the NNs to determine the PI gain constants for these control loops it was necessary to use the data from a 24 – hour simulation run. For this purpose, the data generated from the base case was used. The base case results correspond to the typical responses subject to typical outdoor temperature profile and

cooling loads. For this reason, random data for these variables was generated and changed every hour and a half, to provide a data set for the training of the NN that has sufficient variation and that will include the dynamic behaviour of the system, necessary for the PI – Neuro controllers. The training data set is the same used for the training of the *tsann* and *twsnn*, depicted in *Figure 4.14*. The initial PI controller gains used for this loop were selected by trial and error and were presented in *Table 3-10*. The nature of the non adaptive NNs to be developed has some common characteristics. All of them should predict the output from the plant for the control variable, and all of them have to provide the gains for the PI controller. Tudoroiu, N. and Zaheeruddin, M. (2004) achieves to predict the output from the plant by using a three layer, [5,2,2], feed forward tapped delay NN, with the difference between the set point and the variable to be controlled as an input and the hidden layer output used in the PID controller as gains. A NN similar in structure to the one given by those authors, was trained online and the PID gains were tuned to obtain good tracking performance with minimum number of neurons to avoid over fitting. *Figure 4.18* depicts the NN architecture from the same reference. In the following sections a series of architectures are developed for each one of the control loops. Since only PI controllers were used, the hidden layer would be used to obtain the two gains that are required for the PI controller and the output layer is used to predict the desired temperature.

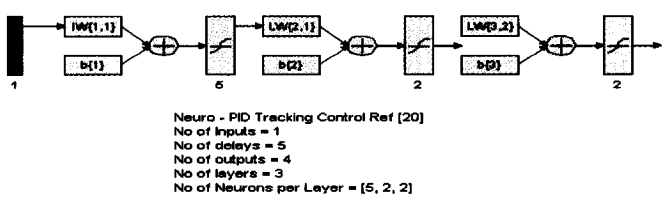


Figure 4.18. Neuro – PID tracking controller Architecture.

4.2.3 Supply Water Temperature Control Loop: *uhpnn*

4.2.3.1 Neural Network Generation and Training Results

After obtaining the training data, several architectures were tried balancing the amount of neurons with the results obtained avoiding over fitting while still getting good prediction and adequate values for the controller gains. The architecture in *Figure 4.19*, a feed back [10, 2, 20, 1] network, with transfer functions for each layer 'purelin', 'logsig', 'purelin', 'purelin' according to the Matlab NN Toolbox was used. A five sample time delay was applied to the three inputs: the heat pump energy input (*uhp*), the supply water temperature (*Tws*) and the corresponding set point (*Twsset*), p = [*uhp; Tws; Twsset*]. The output from the second layer was used to provide the controller with the proportional and integral gains, while the output layer was trained to follow the difference between the water supply temperature and its set point. Since the output of the NN is the input for the controller, the NN is considered to be inverse network. This type of NN was also used by Huaguang, Z. and Cai, L. (2002). *Figure 4.20* shows the comparison of the difference between *Tws* and *Twsset* obtained by the NN and the one from the training data set.

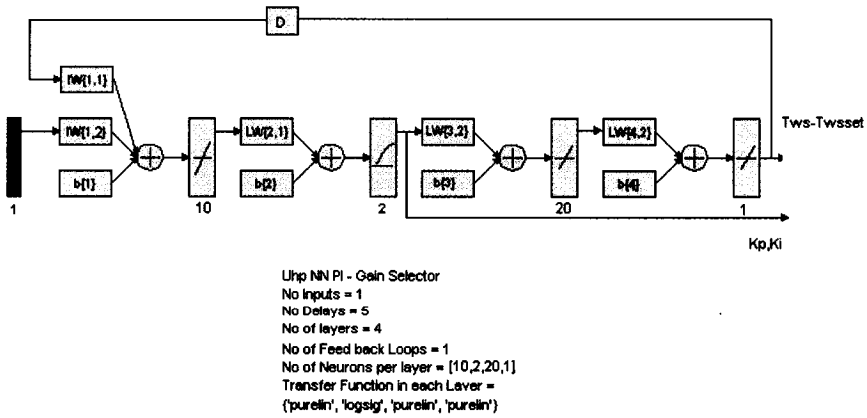


Figure 4.19. Neuro – PID Tracking Controller Architecture uhpnn.

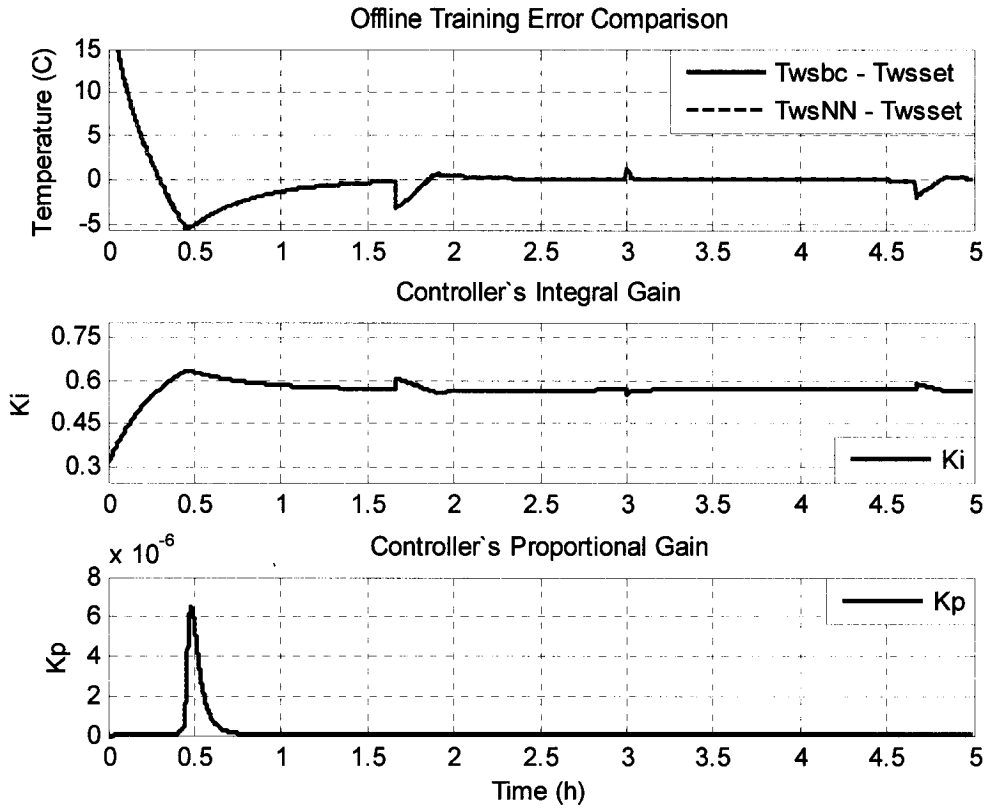


Figure 4.20. Results for Supervised Training for the Supply Water Temperature Loop.

The lower part of *Figure 4.20* depicts the proportional and integral gains of the controller that are given by the hidden layer of the *uhpnn* NN. The integral gain tends to be very small. This type of behaviour was found in most of the architectures tried for this control loop. Consequently, simulation runs were made with the integral gain value fixed to the one from the base case.

4.2.3.2 Response of *uhpnn* to an Unseen Data Set.

Following the supervised training of *uhpnn*, further testing was made with an unseen data set, also using random input temperatures, cooling loads and set points. *Figure 4.21* shows a comparison between the controller with the constant PI gains, the PI-Neuro controller and the PI-Neuro controller with fixed *ki*.

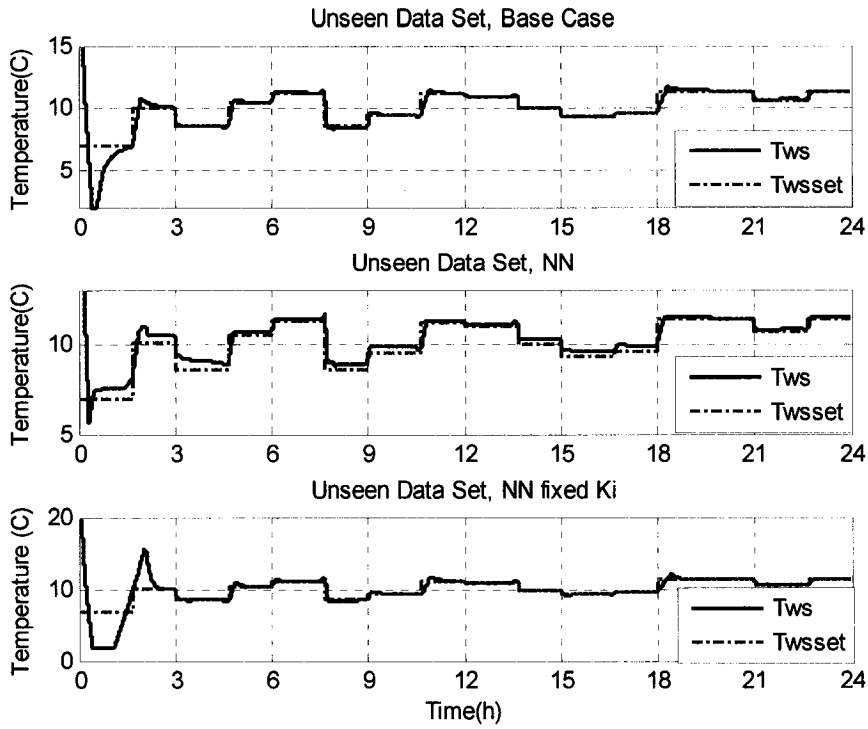


Figure 4.21. Comparison for the Supply Water Temperature Loop with Unseen Data Set.

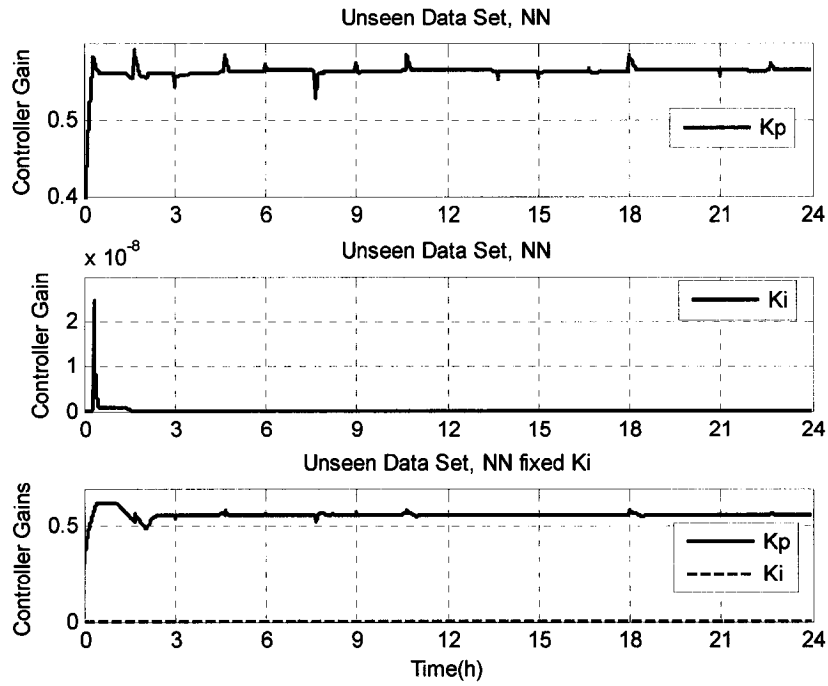


Figure 4.22. Controller Gains for the uhpnn and uhpnn with Fixed ki with Unseen Data Set

It is clear from the Figure (*Figure 4.21*) that the controller reaches the desired set point in each of the different conditions with a small tracking error. However this error is for most of the 24 hours underneath a tolerable 0.5 C. To avoid unnecessary mechanical operation of actuators a higher sample time could be applied. This not only lowers the computational cost, but also proves to be beneficial when encountering real systems, due to normal sampling times in the sensors.

Figure 4.22 shows the corresponding integral and proportional controller gains for the 24h simulation run with the unseen data set, for both the NN and the Fixed *ki* cases.

As mentioned above, other architectures and training procedures were tried, without great improvements in the obtained results. Nevertheless superior data was obtained by just keeping one of the gains constant. The lower part of *Figure 4.21* shows results given by the model just changing the proportional gain by means of *uhpnn*. The integral gain used in this simulation run, was the same one that was determined by trial and error for the base case model (*kiuhp* = 0.0009). In the upper part of *Figure 4.22* is easy to notice that the integral controller gain tend to stay constant although the conditions of the simulation are continuously changing. This conduct of the controller gains was observed in all the architectures developed, and for all the PI controller loops. In the lower part of *Figure 4.22* the controller gains for the fixed *ki* case are displayed.

In the final section of this chapter a comparison between 24 hour simulation runs is done. A system with both controller gains changed by the NN and a system with just the proportional gain changed by the NN are shown. For the purposes of the following sections, the proportional and integral gains in the controller will be changed by the corresponding NNs.

4.2.4 Zone Temperature Control Loop

4.2.4.1 Neural Network Generation and Training Results

For the two Zone temperatures control loops the same architecture and training was used due to the similar operation conditions. In *Figure 4.23*, a feed forward [5, 2, 1] NN, with transfer functions for each layer 'purelin', 'logsig', 'purelin' according to the Matlab NN Toolbox is depicted. Three delays were used for the five inputs: the supply air temperature (*Tsa*), the zone temperature (*Tz*), the outdoor air temperature (*To*), the corresponding zone set point (*Tzset*) and the mass flow rate of air entering the zone (*ma*), pz1 = [*Tsa;Tz1; To; Tz1set; ma1*] and pz2 = [*Tsa;Tz2; To; Tz2set; ma2*], were required to achieve good results. The output from the hidden layer was used to provide the controller with the proportional and integral gains, while the output layer was trained to predict the zone temperature.

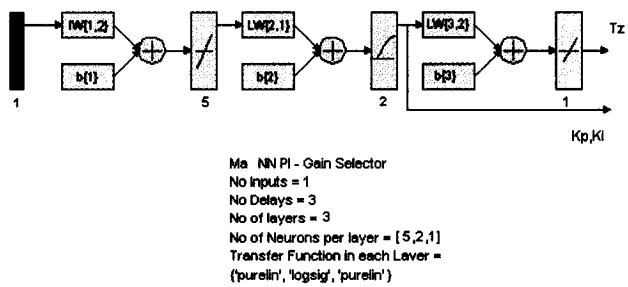


Figure 4.23. Zone Temperature Loops Architecture.

Since the number of layers and neurons for these loops is considerably lower, the training time is lower than that of the supply water temperature loop. Nevertheless in these loops the dynamics of the equations in the model are simpler, so also the requirements in the NNs are lower. *Figure 4.24* depict the results for the supervised training of the architecture with data from zone one while the lower part shows the related gain values

given by the NN under these conditions. The NN prediction from the supervised training revealed some small differences. The maximum error encountered was 0.36 C and the average error was of 0.1 C. Furthermore the results for the controller gains from the NN also show the same trend as the ones observed in the water supply control loop. The integral gain tends to show a constant behaviour, while the proportional controller gain adjusts to modulate the system.

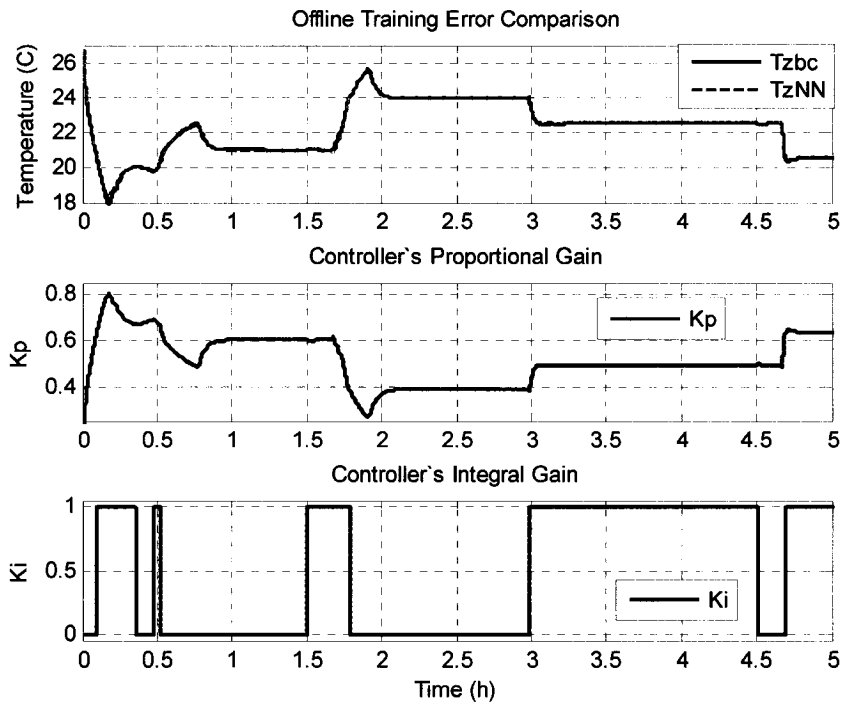


Figure 4.24. Results for Supervised Training for Tz1nn

4.2.4.2 Response of *ma1nn* and *ma2nn*.

In this section, results from the combined NNs for the zone temperature control loops are shown. Since both NNs have the same architecture, and used the same training data the expected results do not vary greatly.

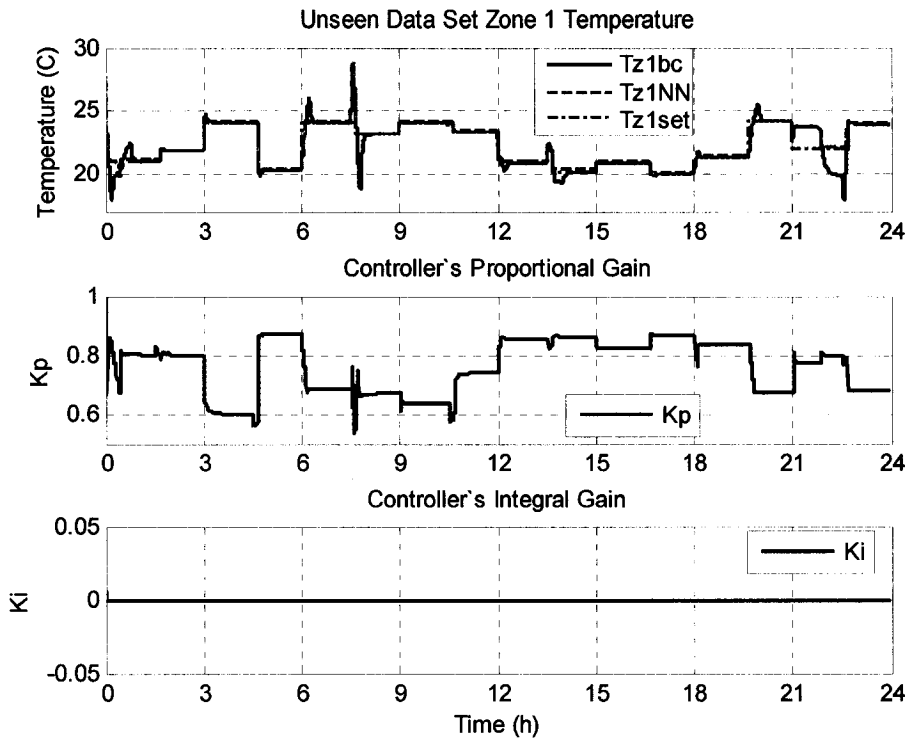


Figure 4.25. Results for the Tz1nn with Unseen Data Set.

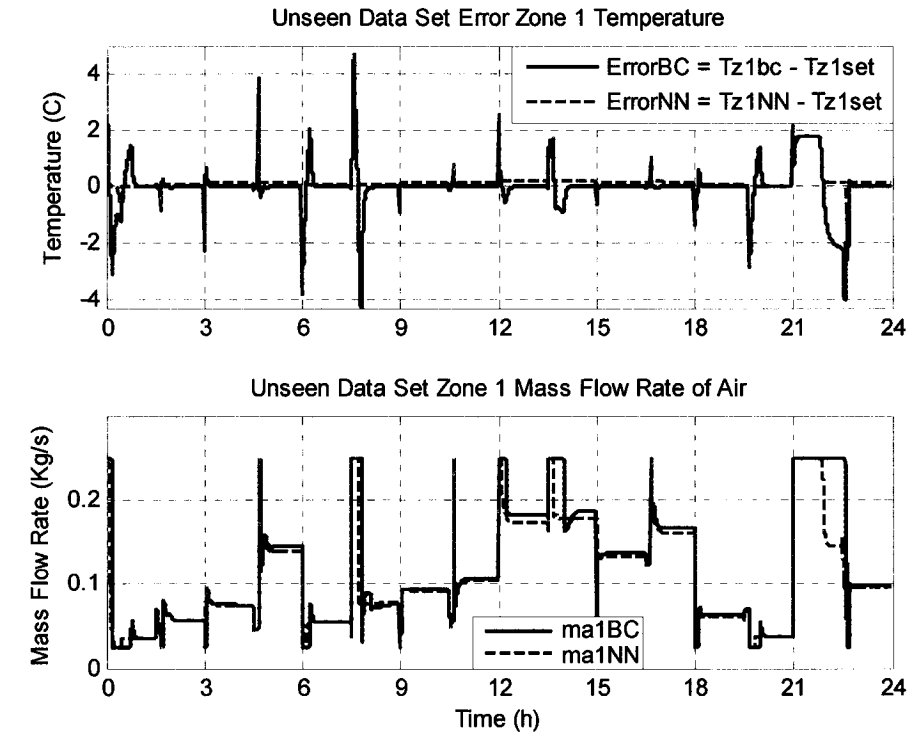


Figure 4.26. Error and Mass Flow Rate of Air for Tz1nn with Unseen Data Set.

Figure 4.25 shows the comparison of the results obtained by the PI-neuro controller in the zone one temperature control loop (*Tz1nn*) with those of the base case. In the lower part of the same figure, the controller gains given by the hidden layer of the NN are displayed. *Figure 4.26* shows the comparison of the error and mass flow rates of air between *Tz1nn* and the Base Case (BC). The error is defined as the difference between the zone temperature and the corresponding set point. The maximum error was 4.64C and the average error was 0.22C. The same simulation run was used to obtain the results depicted *Figure 4.27*, which shows the comparison of the zone 2 temperature between the BC and the NN. The same figure also displays the controller gains selected by the *Tz2nn.* The error and the mass flow rates of air comparisons between NN and the BC are shown in *Figure 4.28.* The maximum error was 10.01C and the average error was 0.29C. It is worth to note that for both zones the NNs do not achieve the required set points. Nonetheless the small steady state error is still in an acceptable range of the required set point. This error is attributed to the integral gain selected by the NNs. Both *Tz1nn* and *Tz2nn* select *ki* to be zero, giving lower undershoots but poor tracking performance. This small steady state error resulted in energy savings in the water supply temperature control loop (the higher the water supply temperature, the less energy consumption in the system), yet is opposite in this case. During the development of the EMC functions in Chapter Three, a penalty functions was included for the time periods in which the zone was not at the required set point, to keep in mind that the energy savings have to be achieved with the minimum discomfort to the occupants of the building. It is suggested to fix the value of *ki*, while still varying *kp* with the NNs, in order to improve the steady state error.

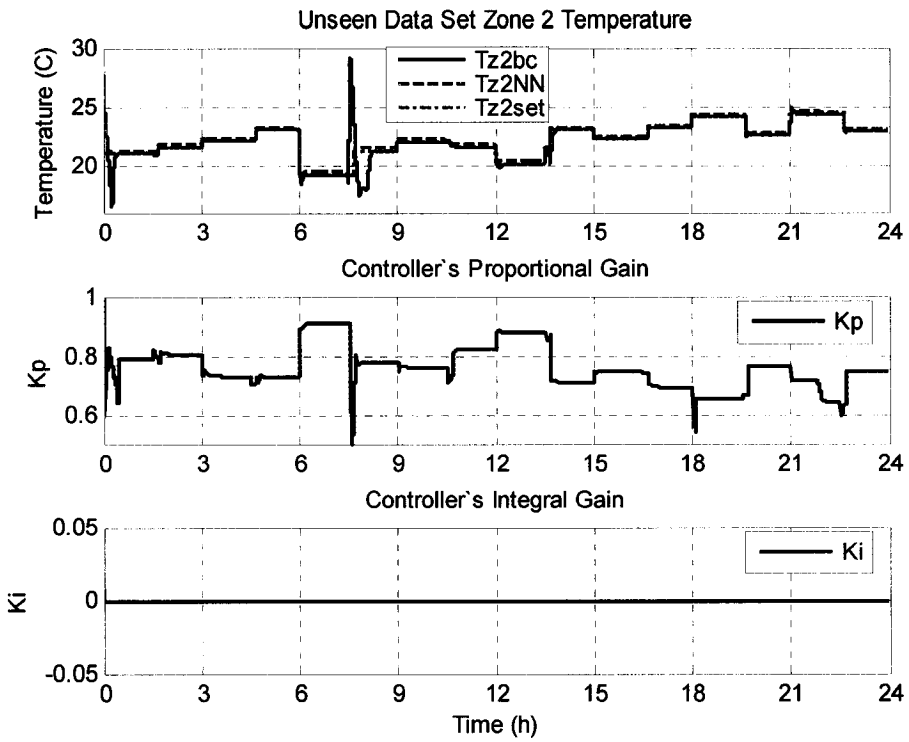


Figure 4.27. Comparison for the Zone 2 Temperature Loop with Unseen Data Set.

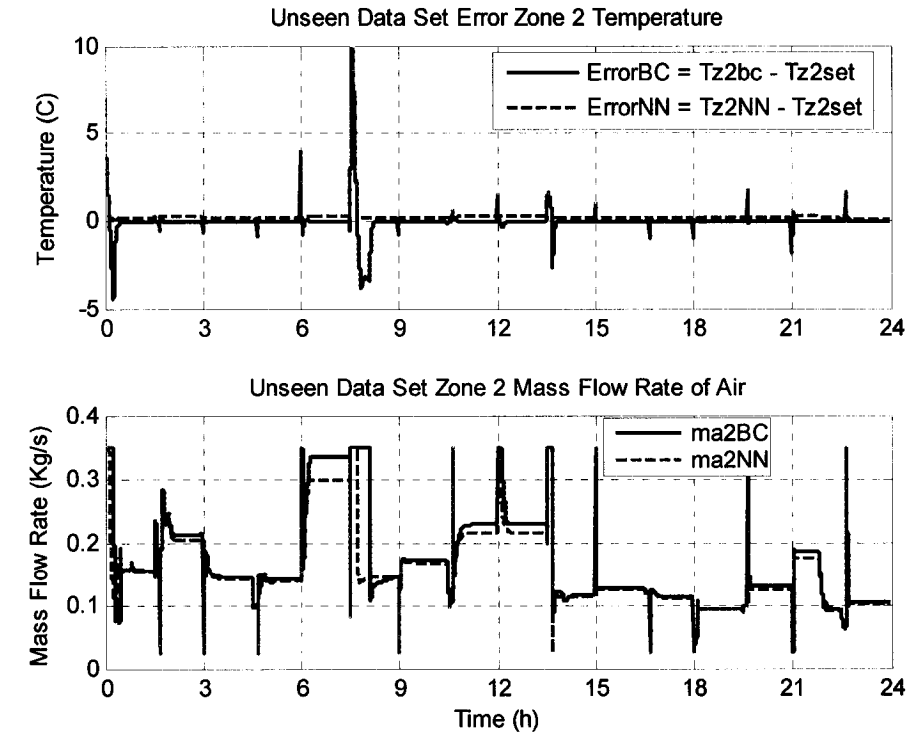


Figure 4.28. Error Between of the Zone 2 Temperature Loop with Unseen Data Set.

4.2.5 Supply Air Temperature Control Loop: *mwnn*

4.2.5.1 Neural Network Generation and Training Results

The neural network architecture for the supply air temperature is shown in *Figure 4.29*. It is a feed back [5, 2, 10, 1] non adaptive NN. The inputs include, the mass flow rate of water (*mw*), the supply air temperature (*Tsa*), the corresponding set point (*Tsaset*) and the supply water temperature (*Tws*), p = [*mw; Tsa; Tsaset; Tws*]. The training time for this particular network turned out to be quite short. The supervised training results along with the controller gain constants given by the NN are displayed in *Figure 4.30*.

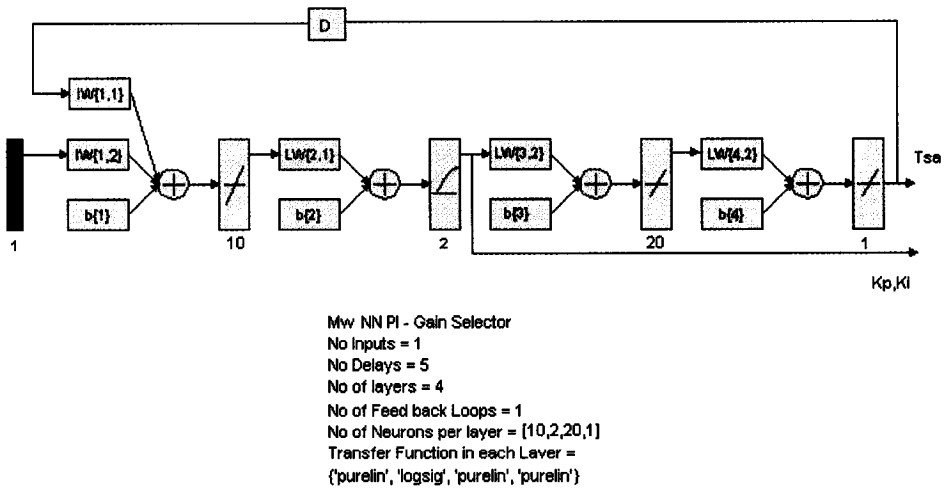


Figure 4.29. Air Supply Temperature Loop Architecture.

4.2.5.2 Response of *mwnn* to an Unseen Data Set.

In this section the results for the supply air temperature control loop are presented. *Figure 4.31* depicts the comparison between the supply air temperature obtained with the NN controller and the Supply air temperature from the BC with an unseen data set.

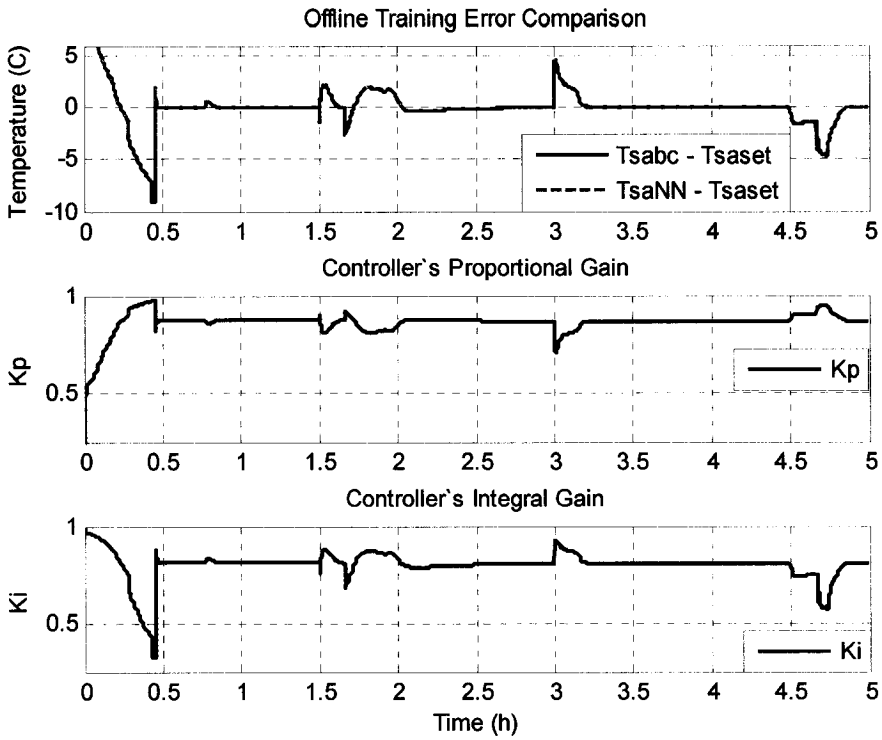


Figure 4.30. Results for Supervised Training for the Air Supply Temperature Loop.

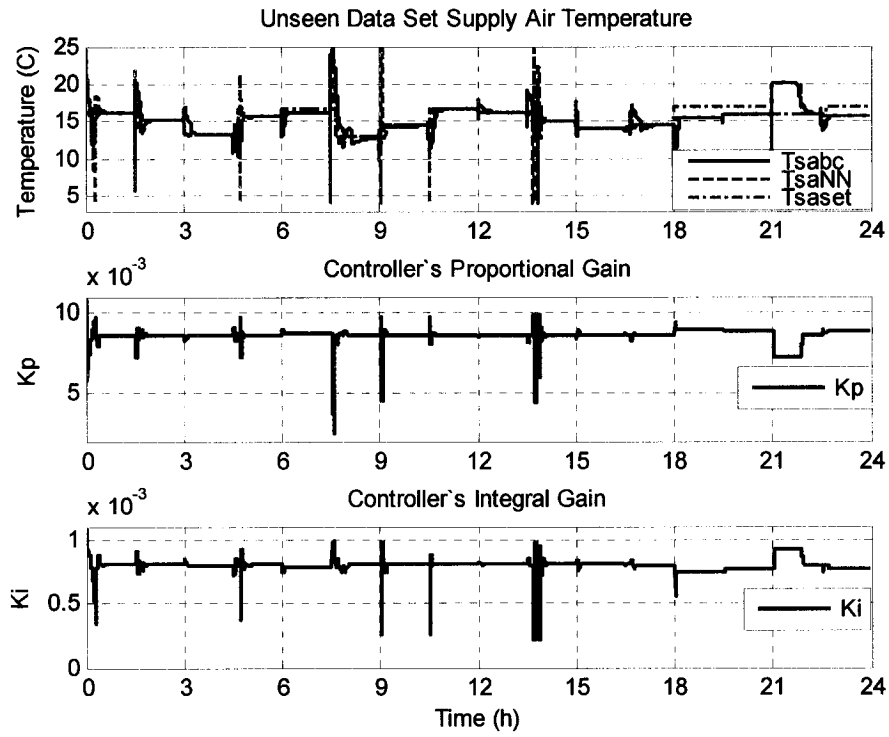


Figure 4.31. Comparison for mwnn with Unseen Data Set.

For this loop, the controller gains are very small, giving a very limited choice for the selection. A scaling factor of 0.01 for *kp* and 0.001 for *ki* was used for the output of the intermediate layer of the non adaptive NN to provide a wider range of operation. In this fashion *mwnn* is able to provide values that vary accordingly to the conditions that the system encounters.

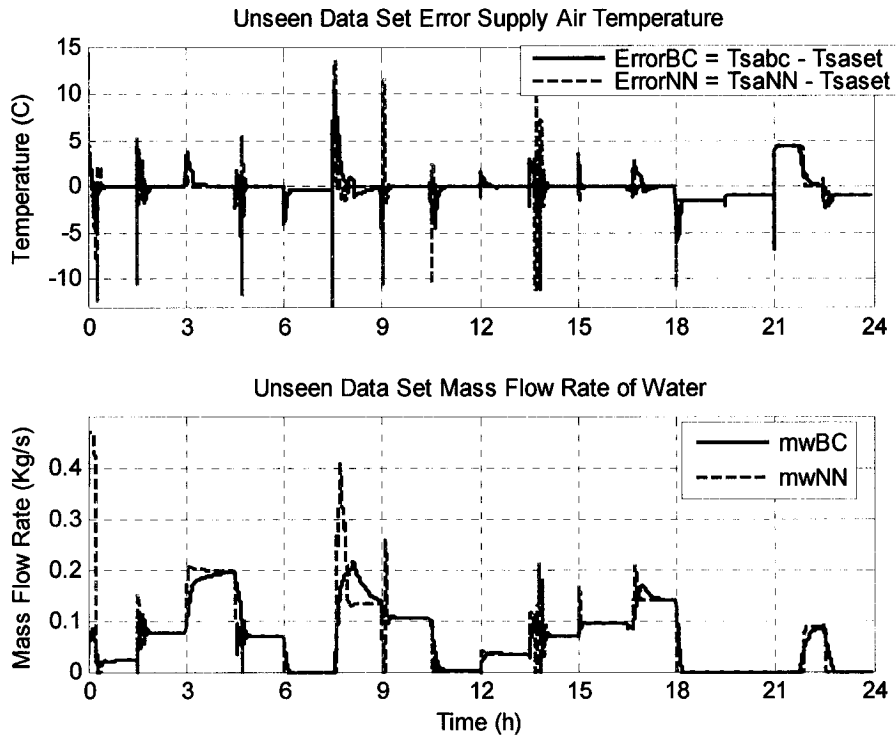


Figure 4.32. Results for the Tsa Loop with Unseen Data Set.

Figure 4.32 shows the error and mass flow rate of water comparison for the BC and the NN. Again the error is considered as the difference between the supply air temperature and its set point. The maximum error was -13.57C very close to the value obtained by the BC, 14.31C. This high error is mainly encountered in the changes between set points. The average error was -0.032C, again close to the value obtain in the BC, 0.029C. The maximum steady state error was -1.458C, for both the NN and the BC, and was during 18h and 19.5h where the mass flow rate of water was 0.

4.2.6 Integrated PI – Neuro – Controllers

This section is intended to compare the base case algorithm, the system with both gains given by the NNs and the system with fixed integral gain and the proportional gain set by the NNs. To start the results for the different control loops for the system with fixed integral controller gain are presented. *Figure 4.33* provides the results for the control loops corresponding to the Zone 1 and Zone 2 temperatures, and the proportional gains for each of these PI controllers. Mass flow rates of air for each zone along with the error between the zone temperature and its correspondent set point are depicted in *Figure 4.34*. Results for the supply water temperature and for the supply air temperature are shown in *Figure 4.35*. Following, *Figure 4.36* depicts the control variables for each one of these control loops and the correspondent errors.

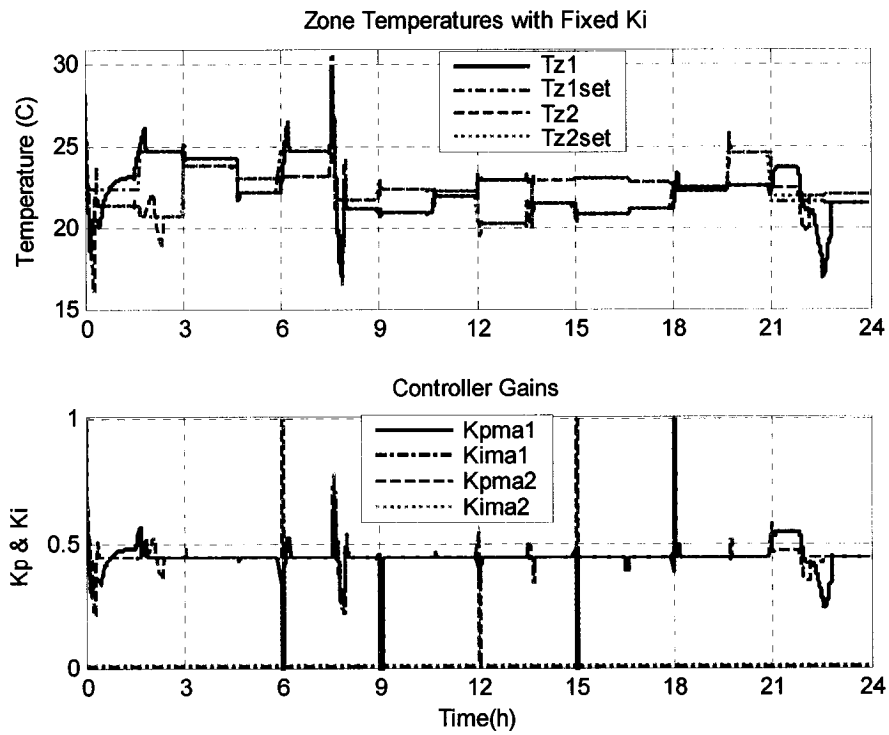


Figure 4.33. Tz1 and Tz2 Loops with Unseen Data Set and Fixed Ki.

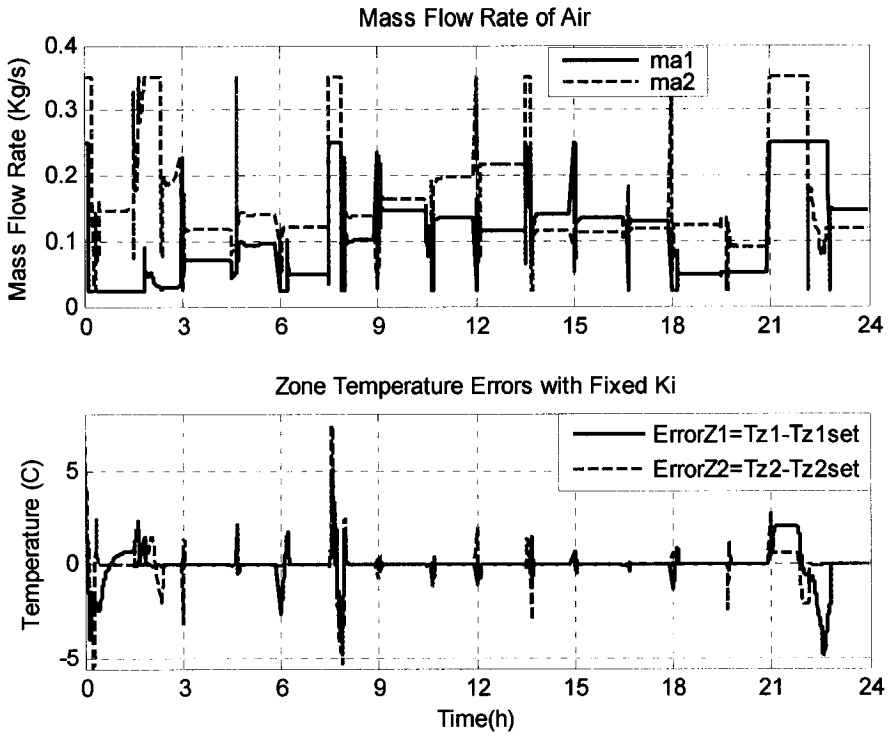


Figure 4.34. Mass Flow Rate of Air and Zone Temperature Errors with Fixed Ki.

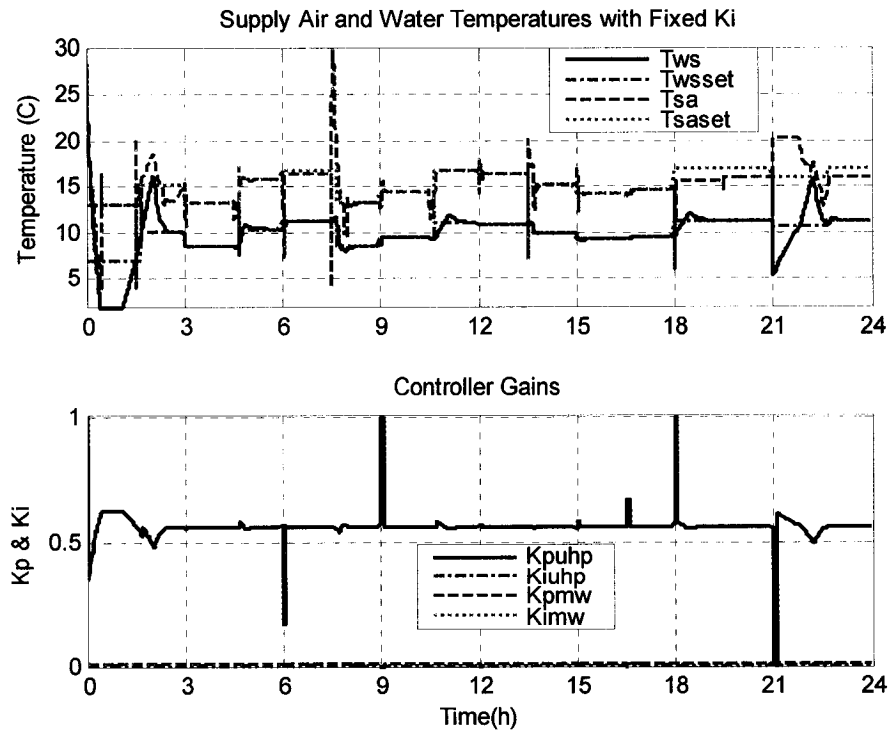


Figure 4.35. Tws and Tsa Loops with Unseen Data Set and Fixed Ki.

	Base Case		PI-NN		P-NN Fixed *ki*	
	Max. (C)	Ave. (C)	Max. (C)	Ave. (C)	Max. (C)	Ave. (C)
Tz1	5.6	0.0	8.2	0.1	5.6	0.01
Tz2	7.1	0.1	7.2	0.1	7.4	0.1
Tws	21.0	-0.5	21.0	0.4	21.0	-0.9
Tsa	17.2	-0.2	13.0	0.3	15.4	-0.4
Energy	240.52 (MJ)		243.93 (MJ)		240.97 (MJ)	

Table 4-10. Error Comparison for the BC, the Neuro- PI and the Fixed ki Cases.

Table 4-10 displays the maximum and average errors for the three cases under the same conditions. It also displays the total energy consumption for each of the cases.

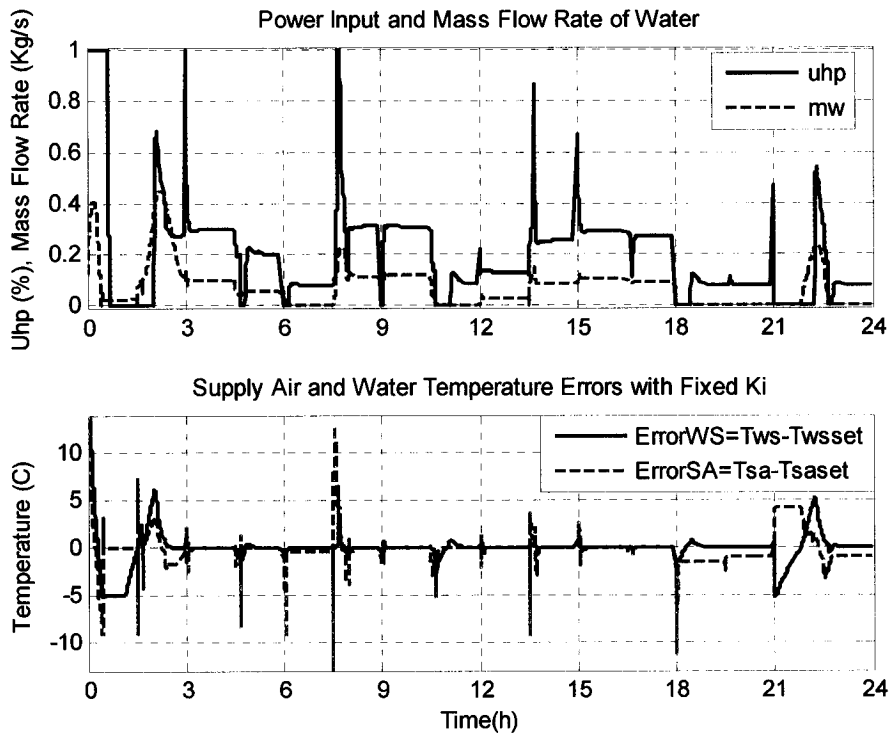


Figure 4.36. Error for the Control Loops with Unseen Data Set and Fixed Ki.

Figure 4.37 shows the comparison between the three strategies: Base Case, PI-Neuro Controller, P-Neuro Controller Fixed *ki*. The total energy consumption given by the three strategies was very close. The results shown in these figures expose the better tracking performance and energy consumption of the proportional-Neuro controllers with fixed integral gain than those of the Neuro-PI. In general, the set points were reached with less

undershoot and consequently in shorter periods of time with the fixed *ki*. Furthermore unnecessary damper operation is avoided. Furthermore this strategy has almost the same energy consumption than that of the BC. For the reasons exposed above the fixed *ki* strategy was selected for the integrated system accounted in the next section.

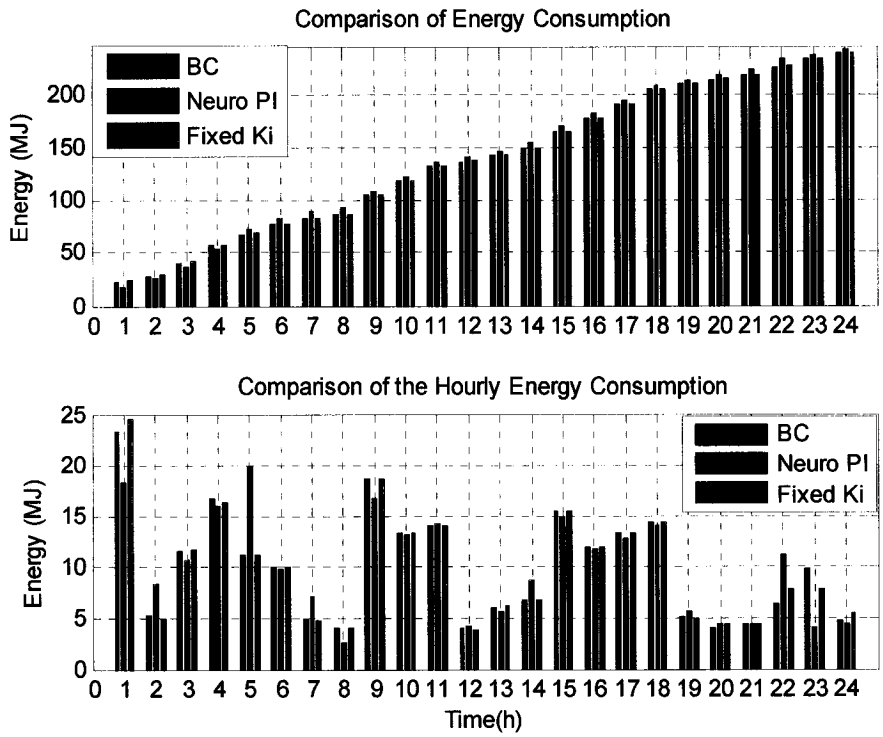


Figure 4.37. Energy Comparison for the BC, PINN, PNN Fixed KI and Combined System.

4.3 Neuro – EMC Integrated System

After the creation and testing of the NNs, it is necessary to combine them and create a single model of the Neuro energy management control system (Neuro – EMC). One by one, each NN was added to create the integrated system to be compared with the BC algorithms. The following section shows the results and improvements of this integrated Neuro- EMC system. *Figure 4.38* shows the typical daily zone temperature profiles obtained with the PI neuro controller holding the integral gain value constant, *Kimal* =

0.007 and *Kima2* = 0.008, which are displayed in the lower part of the figure shows the corresponding controller gains for each of the zones. From this figure is clear that the set points for each of the zones is reached and maintained during the occupied period. The slight variation from the set point observed between five and six in the afternoon (17h and 18 h) is due to the stopping of the heat pump given by the stop lead time algorithm. *Figure 4.39* shows the mass flow rates for each zone compared to those of the BC; the lower part of the same figure displays the outdoor air intake for the system, *xv*, given by the NN compared to the one by the BC. As expected the NN follows the required behaviour for the outdoor air economy cycle. *Figure 4.40* shows in the upper part the comparison of the supply air temperature and its corresponding set point for both the fixed *ki* and the BC. The lower part displays the controller gains given by the NN for this loop. *Figure 4.41* shows the comparison of the mass flow rate of water (*mw*) and power intake (*uhp*) for the NN controlled system and the BC. Since the set point generated by the *tsann,* from section 4.1.3, is lower by 0.1304C (in average) from that of the BC, the mass flow rate of water of the system is increased during the operation. From this figure is also clear that the start time given by the NN is earlier than that of the BC algorithm. *Figure 4.42* depicts the comparison between the NN and the BC for the supply water temperature control loop, and Temperature reset algorithm. The lower part of the figure shows the controller gains selected by the NN. The difference between the generated set point by *twsnn,* from section 4.1.3 and the BC algorithm was 0.0263C, which is smaller than the one for the *Tsa* loop. However the steady state error was higher, as shown in both *Table 4-11* and *Figure 4.42*.

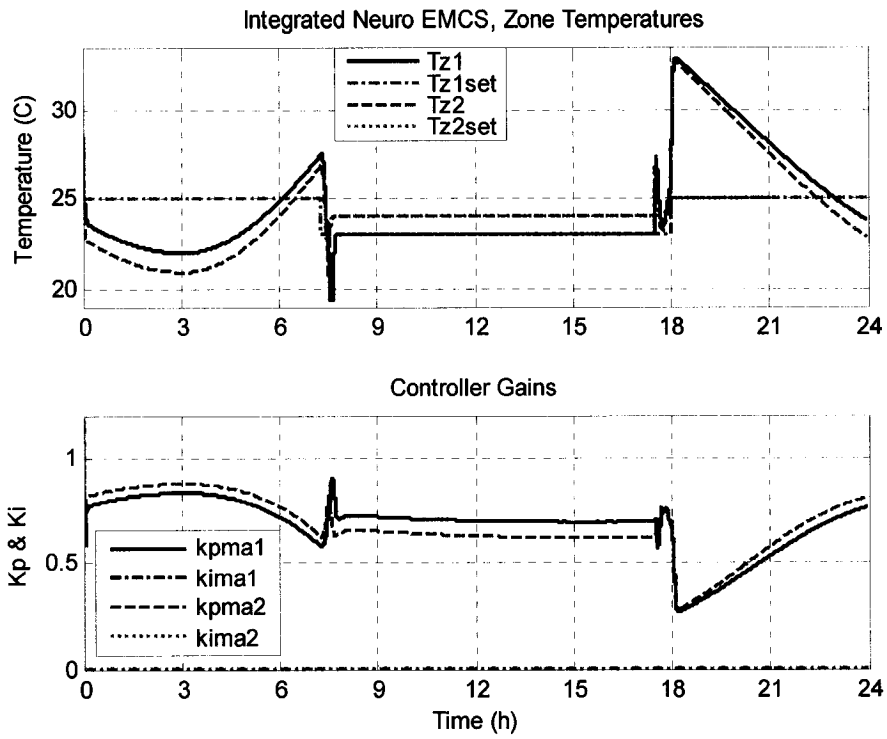


Figure 4.38. Zone Temperature Profiles with Fixed KI.

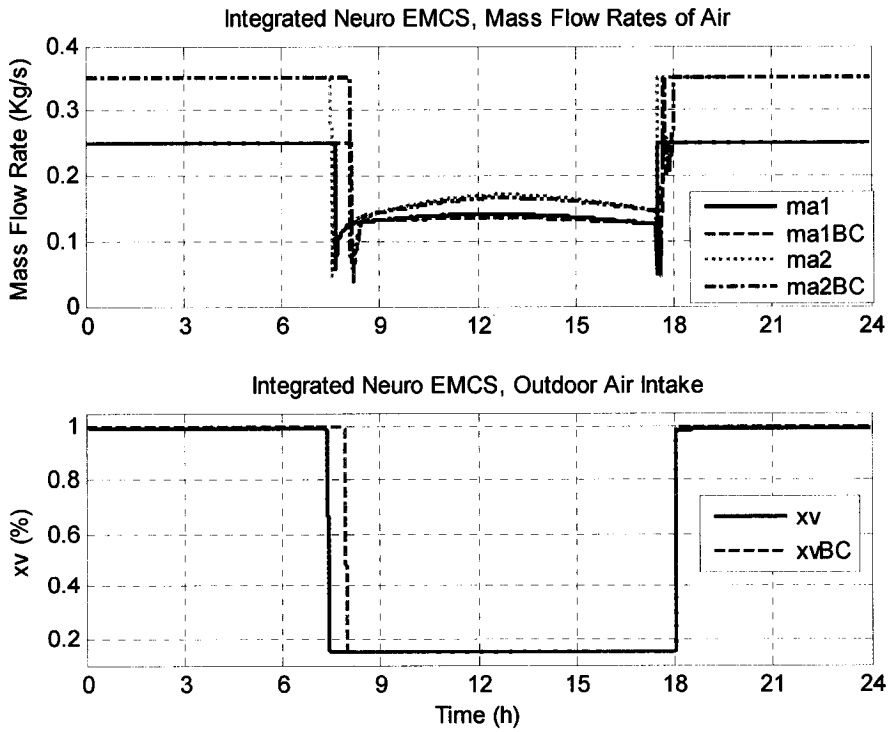


Figure 4.39. Comparison of the Mass Flow Rates of Air and Indoor Air Intake.

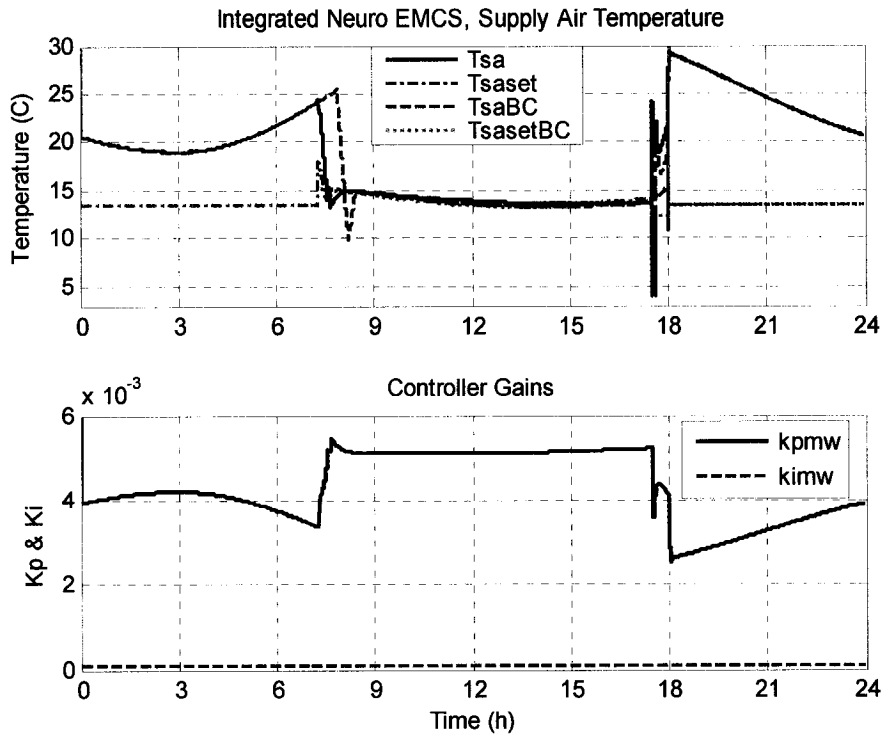


Figure 4.40. Comparison of the Supply Air Temperature Control loop and Reset Strategy.

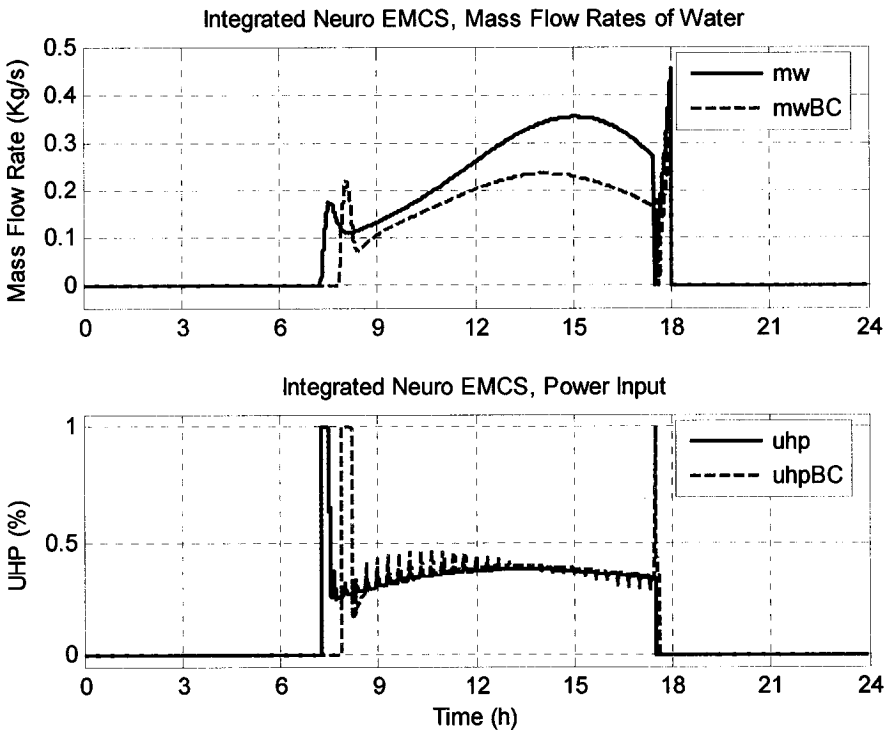


Figure 4.41. Comparison of the Mass Flow Rate of Water and Power Input.

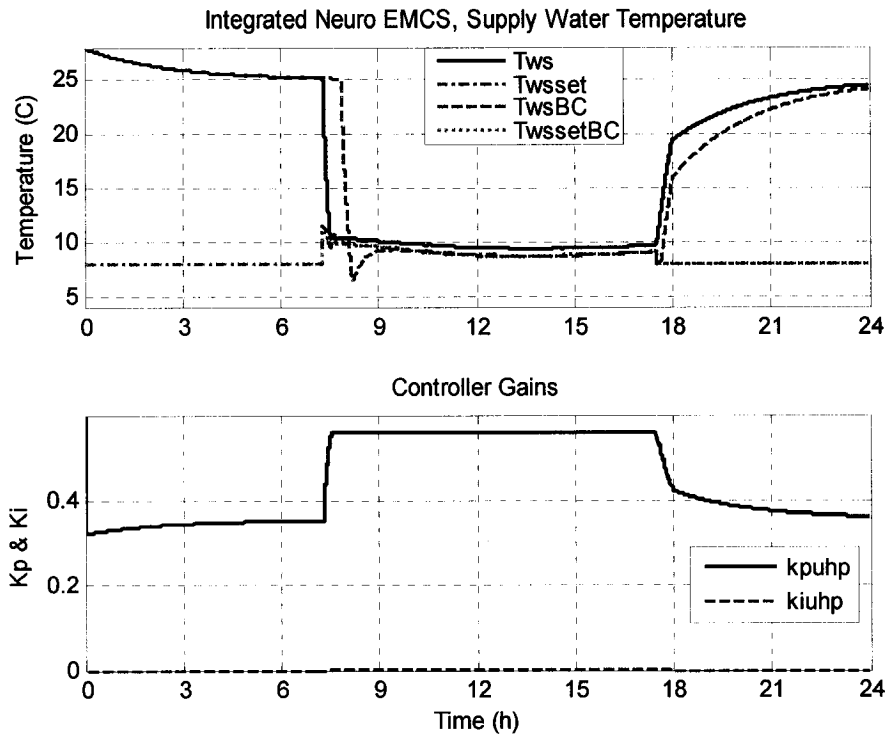


Figure 4.42. Comparison Supply Water Temperature Control loop.

The steady state error between the supply water temperature and its set point partly explains the error encountered in the supply air temperature loop. However, since the zone temperatures reached their set points the temperature regulation is not compromised. A set of adaptive NNs is recommended in order to improve tracking performance.

	Base Case		P-NN Fixed *ki*	
	Max. (C)	Ave. (C)	Max. (C)	Ave. (C)
Tz1 - Tz1set	-5.1	0.5	-3.8	0.4
Tz2 - Tz2set	-4.1	-0.1	-4.1	-0.1
Tws - Twsset	-3.4	0.1	0.4	0.7
Tsa - Tsaset	-8.3	0.2	-10.1	0.2
Start Time	28302 (s)		26272 (s)	
Stop Time	63439 (s)		62943 (s)	
Energy	219.1623 (MJ)		219.9697 (MJ)	

Table 4-11. Error Comparison for the BC and the Fixed ki Cases.

Table 4-11 depicts the maximum and average errors for both the BC and the NN as well as the start and stop times and total energy consumption. From these results it is evident

that the group of NN perform as well as the algorithms. The undershoots are lower and reach the set points within the tolerable errors. It is apparent that the energy consumption of the group of NNs is slightly greater that that of the BC. This is due to the earlier start time that translates in more operational time, therefore more energy is consumed. However this difference is compensated with an earlier stop time and with lower energy consumption during operation due to the lack of tracking performance of the PI-Neuro controller for the supply water temperature control loop. *Figure 4.43* shows a close up of the start and stop times with the zone temperatures. As mentioned before, the start time given by the NN is 26,272 seconds, equivalent to 7:18 am, while the start time obtained by the BC algorithm was 28,302 seconds, equal to 7:52 am. However the zone temperatures from the BC are not in the desired range (plus or minus 2C form the occupied set point). The selected time from the NN ensured the appropriate temperature during the occupied time. Further training can be made to the NN to diminish the extra twelve minutes gained due to the faster convergence time of the PI- neuro controllers. This could be achieved by means of an adaptive NN. For the stop time, both the base case and the Neuro EMC system, were underneath the acceptable threshold for the zone temperature during the occupied time (plus or minus 2C from the set point). The NN predicted a stop time of 62,943 seconds equivalent to 5:29 pm while the base case stop algorithm prediction was 63,439 seconds equal to 5:37 pm. This is translated in higher energy savings or the Neuro EMC system. However, it is worth to note that further training is recommended, in order to accommodate to the PI- Neuro controllers, and avoid the peaks displayed in the figure as soon as the heat pump is turned off (see *Figure 4.43* between 17.4 and 17.6).

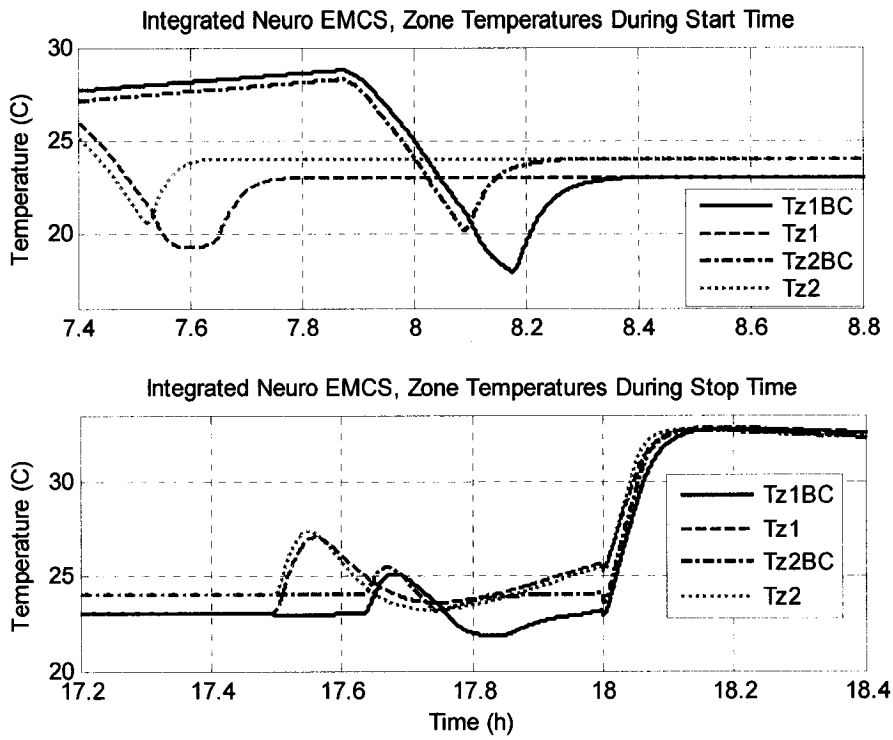


Figure 4.43. Comparison of the Start and Stop Lead Times from the NN and the BC.

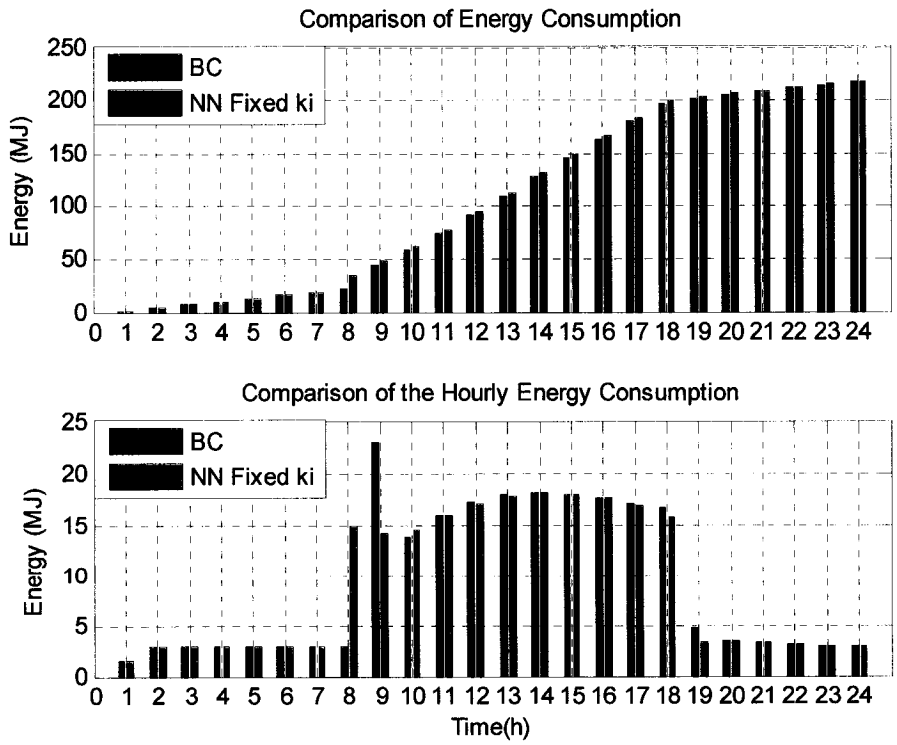


Figure 4.44. Comparison of the Energy Consumption of the NN and the BC.

In summary *Figure 4.44* depicts the comparison of the energy consumption between the Neuro EMC System and the BC algorithms. It is clear that the energy consumption of the system with the NNs is a bit higher than the one from the base case (0.81KJ) a merely 0.4%with a maximum of 35.2% when the NN system started. If the hourly energy consumption is compared, it is evident that during normal operation the system would consume an average 1.2% less energy than the base case. As mentioned above the total energy consumption was mainly altered by the difference in start times, nevertheless the results obtained with the neural networks appear to be reliable.

Further training is required to improve the performance of the Start and Stop lead time NNs. Also for the selection of the PI- Controller gains is recommended a set of adaptive NN to improve tracking performance.

4.4 Conclusions

1. The NNs are able to predict adequate start and stop lead times, without compromising the temperature comfort of the occupants of the building.
2. The outdoor air economy cycle NN displays good performance, and a relative low time of training. In the comparison of the energy consumption of the base case outdoor air economy cycle algorithm and the NN, up to 80KJ were saved during a 24h simulation run. Further savings can be achieved, by forcing the NN to reach maximum outdoor air intake.
3. The most common algorithms for the reset temperature strategy require the knowledge or measurement of the heating or cooling load in the zones. Using data available, the system dynamics are captured by the NNs obtaining adequate set

points to reach the desired zone temperature while adapting to the constant changing conditions in the system. Energy saving between the BC Algorithm and the NN were not significant.

4. The PI- Neuro controllers developed prove to have good prediction qualities, but somewhat lacking in the tracking performance. From the results it was clear that the integral controller gain generated by these NNs was maintained almost constant throughout the whole simulation runs. Subsequently, a combined system with the proportional gain selected by the NN and the integral gain remaining fixed was developed, obtaining good tracking of the selected set points.
5. Energy consumption of the integrated Neuro EMC system proves to be slightly higher than the one form the BC. This was due to an earlier start time given by the Start time NN. Nevertheless when the hourly consumption of the two systems is compared, and the effects from this initial greater energy consumption are isolated, the Neuro EMC slightly reduces the energy consumption on an average of 1.2% per hour.
6. Is recommended to create a set of Adaptive NNs. This could improve the tracking performance and the overall energy consumption of the system.

5 ADAPTIVE ARTIFICIAL NEURAL NETWORKS

5.1 Introduction

The aim in this chapter is to develop a group of adaptive NNs, to replace those created in Chapter 4. This adaptive Neuro – Energy Management Control System (EMCS) will be compared to both the base case strategies, and the non-adaptive Neuro –EMCS with respect to energy consumption and tracking performance. At the end of the chapter a hybrid system, composed of the best performing EMS or Neuro-EMS algorithms will be presented.

5.2 Adaptive Neural – EMC Function

5.2.1 Introduction

In the previous Chapter a series of non adaptive NNs were developed to facilitate the implementation of energy management strategies in buildings while also improving the overall performance of the HVAC system. Even though the NN performed well, there is room for improvement. To this end, a group of adaptive NNs is suggested to fully exploit the advantage of generalization and prediction of NN.

5.2.2 Start Stop Lead Time Adaptive Neural Networks

As explained in Section 4.1.1, the selection of a start or stop lead time is highly dependent in the room conditions, occupation, outdoor temperature, system dynamics, etc. These factors change from day to day, requiring as well changes in the lead times. With the purpose of improving the performance of the already existing non adaptive NN an

algorithm to adapt the existent NN is implemented. During the first 10 days the algorithm is in place, data is collected to ensure a good training set. After this time, the system collects data every 3 days to continue the training process based on the data collected during the 10 precedent days. This is expected to adapt the NN to the ongoing changes in the zones in order to provide better start and stop lead times for the HVAC system. *Table 5-1* displays the input and the desired output for training the start lead time NN collected in 10 days of simulations, while *Table 5-2* shows the data used for the stop lead time NN.

Day	***To*** (C)	***Tsa*** (C)	***Tz*** (C)	***Tzset*** (C)	**Lead time** (s)
1	23.6	23.5	26.8	23.0	1340
2	20.8	20.8	24.1	21.0	1250
3	21.0	21.0	23.1	23.0	1355
4	20.4	20.4	22.7	22.0	1593
5	21.5	21.5	23.8	25.0	1171
6	22.0	21.9	23.9	22.5	825
7	21.6	21.6	23.9	21.5	1080
8	19.0	19.0	21.3	22.4	1003
9	22.2	22.3	24.2	21.5	1469
10	21.7	21.8	24.0	23.5	991

Table 5-1. Collected Data for the Online Training of the Start Lead Time NN

After this data was collected, the start and stop lead times of the newly adapted NN are compared with those obtained by the BC algorithm and from the non adaptive NN. The plant used for each of the cases has constant PI controllers with the purpose of comparing the results under similar operating conditions. Later on in the chapter the other NNs will be combined. *Figure 5.1* displays the temperature profiles obtained after 11 days of consecutive simulation for zone one, during the start and stop time, while *Figure 5.2* displays the results for zone 2. *Table 5-3* shows the correspondent start and stop lead times for each strategy.

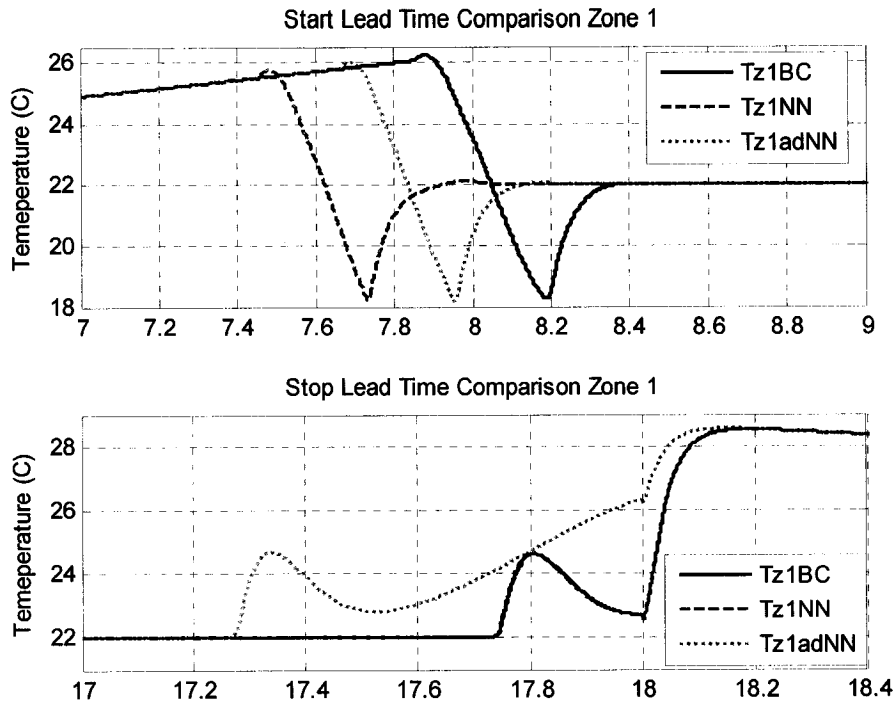


Figure 5.1. Start and Stop Lead Time Comparison for the BC, NN, Adaptive NN, Zone 1.

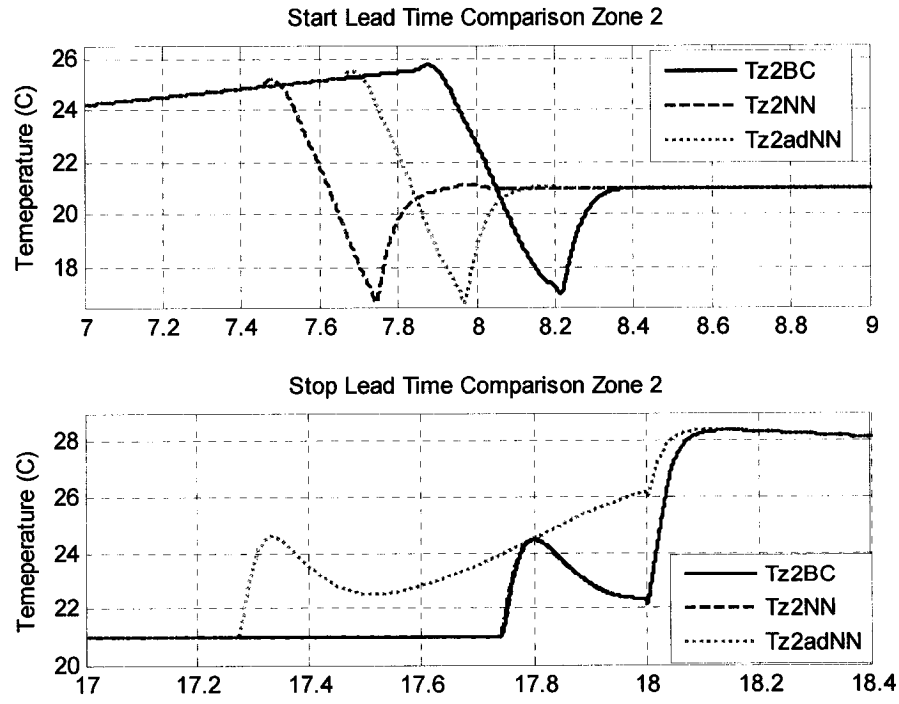


Figure 5.2. Start and Stop Lead Time Comparison for the BC, NN, Adaptive NN, Zone 2.

Day	***To*** (C)	***Tsa*** (C)	***Tz*** (C)	**Lead time** (s)
1	30.6	13.0	22.0	3177.0
2	23.5	14.9	23.0	3433.0
3	22.1	15.6	23.8	3457.0
4	16.4	16.4	24.2	3420.3
5	16.4	15.7	21.2	3590.0
6	33.4	12.1	21.2	1980.5
7	27.5	13.9	22.3	2610.3
8	24.7	14.5	22.3	2880.4
9	26.7	14.9	24.5	2700.8
10	33.4	13.2	23.5	1980.3

Table 5-2. Collected Data for the Online Training of the Stop Lead Time NN

Strategy	Start Time (h)	Error at 8h		Stop Time (h)	Error at 18h	
		Z1	Z2		Z1	Z2
Base Case	7.9	1.7	1.8	17.7	0.7	1.4
NN	7.4	0.1	0	17.7	0.7	1.4
Adaptive NN	7.7	1.6	2.3	17.5	2.6	3.5

Table 5-3. Comparison of the Start and Stop Lead Times after Training

Figure 5.3 shows the comparison of the total and hourly energy consumption for the 24h simulation run. The maximum energy consumption was 222 MJ, for the NN system, followed by the BC with 218MJ. As expected the performance of the adaptive NN is slightly better with 216 MJ. It is clear that the lowest energy consumption during the start time is going to be the one of the BC, due to the late start time. However, as in chapter four, the BC is not at the acceptable range of temperatures (*Tzset* ± 2) at 8am, when the building is occupied, reaching the proper range 12 minutes later and the desired set point at 8:23am.

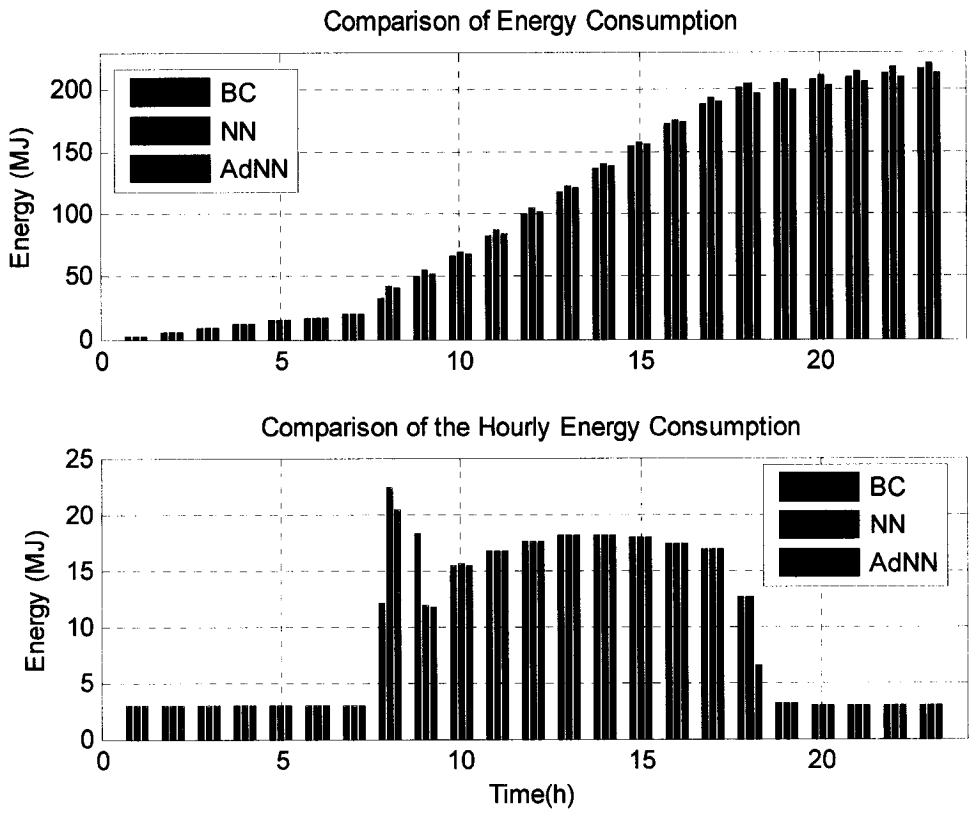


Figure 5.3. Hourly Energy Consumption for the BC, NN, Adaptive NN.

The adaptive NN (adNN) displays a better performance over the NN, starting 16 minutes later, being in the range of acceptable temperatures point at 8am and reaching the set point at 8:06am. The non adaptive NN consumed 15.2% more energy than the BC, while the adNN consumed 6.4% more energy than the BC in the start case. For the stop time the adNN strategy stopped the heat pump before the other two strategies, producing savings of 15.5% in the energy consumption in respect to both the BC and the NN strategies. Yet, the temperature of the zones is outside of the acceptable range at the time when the building becomes vacant, 6:00 pm, hence the results from the adNN are expected to improve with further training. It is suggested to keep a larger training data set, of 80 to 90 days, to keep a wider record of the building and outdoor temperature

behaviour. Nevertheless the training can still be made every 3 days, this will employ in a better way the generalization capabilities of the NNs.

5.2.3 Reset Supply Air Temperature and Water Supply Temperature Adaptive Neural Network

In the case of the reset water and air supply temperature, the development of the adNNs is not directly based on the architecture showed in section 4.1.3. This is due to the necessity of including more parameters to allow the adNN to learn from its own results. The first thing that has to be bear in mind is that the modeled system has two zones, with independent set points and only one coil to provide the supply air temperature to satisfy the zones' cooling or heating requirements. The set point selection has to provide enough cooling or heating to satisfy both zones, while maintaining the water supply set point as high as possible, to diminish the energy consumption of the heat pump. The second point is the time selection for the data acquisition for the training. It is necessary to provide an initial online training to ensure that the adNNs will provide results that are logical. This point will prove crucial when combining the strategy with the adNNs for the PI controllers developed in later sections. Equation (36) describes the algorithm used during the first 5400s for the online training data.

$$\begin{aligned} e &= Tz - Tzset \\ r &= random(0,0.5) \\ Tset &= \begin{Bmatrix} Tset = Tset + r & e \le 0.5 \; or \; ma \le 0.3 \\ Tset = Tset - r & e > 0.5 \end{Bmatrix} \end{aligned} \tag{36}$$

This simple algorithm has the purpose of teaching the adNNs the desired direction (increase or decrease) of the set point.

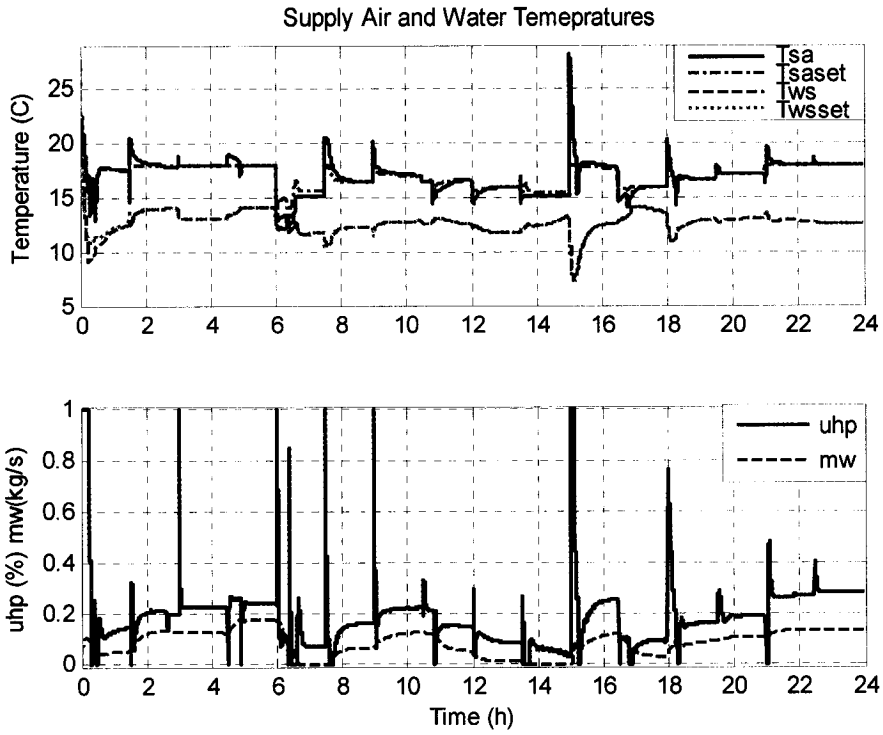


Figure 5.4.Supply Air and Water Temperatures and mw and uhp

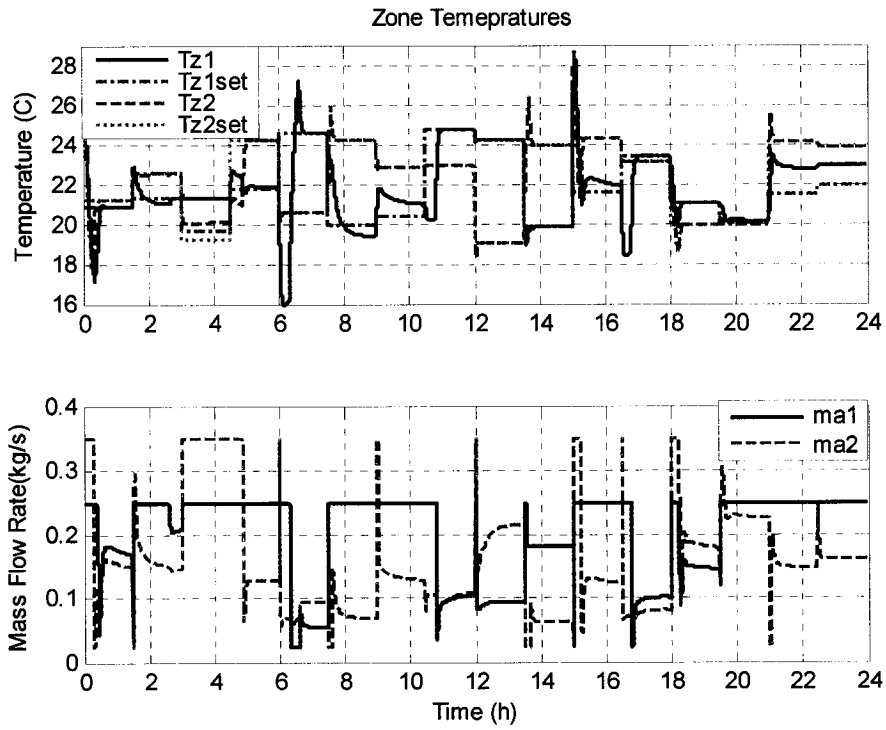


Figure 5.5. Zone Temperature and Mass Flow Rate of Air.

This will just allow the outputs from the NN to be in a desired range and it is expected that the generalization capabilities of the adNN will use the random selected numbers as a guide and improves the results over time. The architecture that displayed the best results for both cases was a feed forward [8, 5, 1] with 'purelin', 'purelin' and 'poslin' as transfer functions. The inputs for the supply air temperature set point was p = [*To*; *Tsa*; *Tz1-Tz1set*; *ma*] and for the supply water temperature set point was p = [*To*; *Tws*; *Tsa-Tsaset*; *mw*]. To avoid inconsistent results a minimum and a maximum was imposed for the set points. *Figure 5.4* displays the air and water supply temperatures and its respective set points given by the adNNs in the upper part, while in the lower part the percentage of power input *Uhp* and the mass flow rate of water *mw* are depicted. *Figure 5.5* shows the zone temperatures and its set points in the upper part while in the lower part the mass flow rates of air are depicted. Another option that was considered was a single adNN, providing both *Tsaset* and *Twsset.* A feed forward, five layer adNN, [8, 4, 5, 1, 1] with 'purelin' in the first three layers and 'poslin' in the last two as transfer function. This adNN displayed similar results as the individual adNN, however the energy consumption was higher than that of the individual adNN (39.62 MJ), rendering no real benefits in combining the two adNN. *Figure 5.6* shows the total energy consumption between the BC, the non adaptive NN and the individual adNNs, while the lower part of the same figure displays the comparison of the hourly energy consumption for the same strategies. The maximum energy consumption is given by the non adaptive NNs, 267.58MJ, followed by the BC with 246.61MJ. The adNNs show the best results with 244.32MJ. Although the difference is not significant, the adNNs proved to improve the performance of the non adaptive ones, with simple rules for the online training.

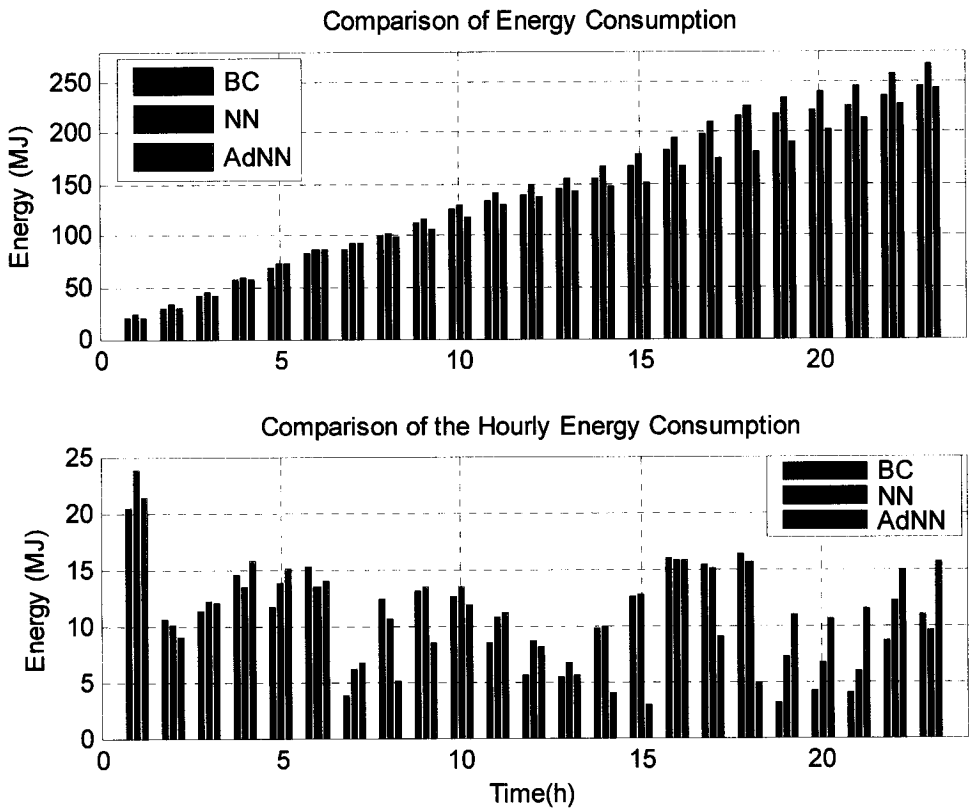


Figure 5.6. Energy Comparison for the BC, NN, Adaptive NN and Single Adaptive NN.

5.3 Adaptive PI – Neural controllers

5.3.1 Introduction

In chapter 4 the performance of the NNs in selecting proper proportional and integral constants for the controllers proved not to be entirely satisfactory. The steady state error encountered in the controller was higher than the one from the controller with the constants selected by trial and error. The main reason for this greater error was the tendency of the NNs to approach the integral constant to zero.

In this section it is our intention to develop a series of adaptive neural networks (adNNs) to enhance performance. To start this section simplified individual loops were used to test

the results of the adNNs in comparison with the BC loop and a non-adaptive NN; later on the adNNs will be combined in order to test the results obtained by the three different systems: Base Case, non-adaptive NNs and adNNs.

5.3.2 Water Supply Temperature Loop

For the water supply temperature control loop *Figure 5.7* shows the behaviour of the PI controller with constants selected by trial and error summer conditions. For this case, maximum mass flow rate of water ($mw = 0.468$) was used; the water return temperature was kept constant at 12C. The set point for the supply water was chosen according the equation (24b) in Chapter 3.

$$Twsset = -0.18 \cdot To + 13.6487 \qquad (24b)$$

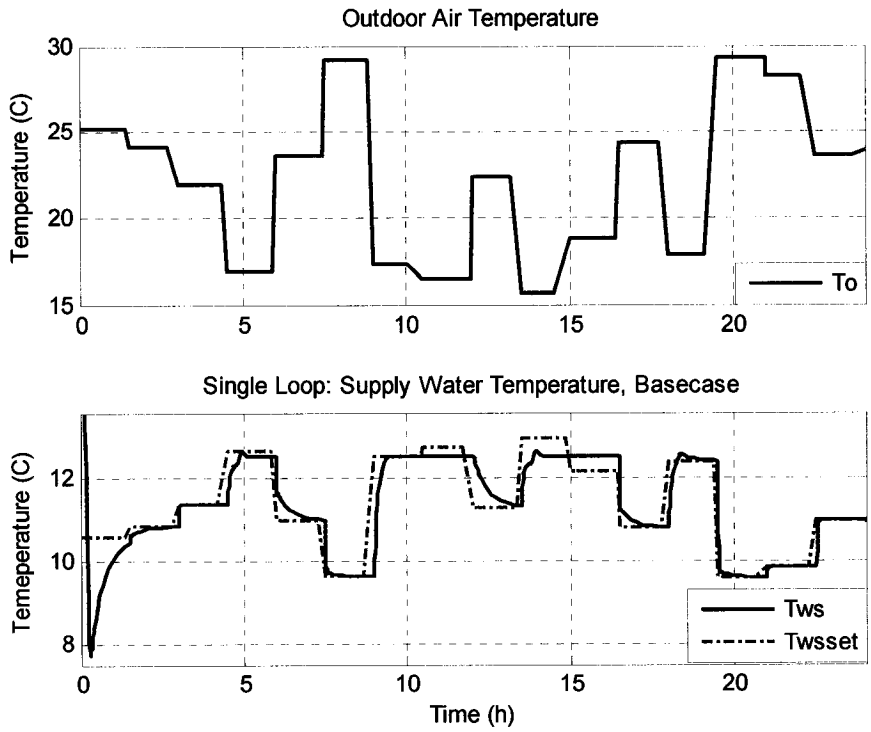


Figure 5.7. Single Loop Simulation for Fixed Kp and Ki.

The set point is in most cases achieved (81% of the cases shown in *Figure 5.7*). *Figure 5.7* displays error for the supply water temperatures. The maximum error for summer simulation was 2.88C while the average error was 0.6789C.

The next step was implementing the non-adaptive NNs developed using the procedures in chapter 4 in the single loop system. Using the data from the single loop base case above a new NN, [5, 2, 15, 1], with transfer functions 'tansig', 'poslin', 'tansig' and 'purelin', in each of the corresponding layers was created. The same three inputs (heat pump energy supply, *uhp*, supply water temperature, *Tws* and the supply water temperature set point, *Twsset*) used in the selected NN from Chapter 4 with five time delays were employed, and again the error between the supply water temperature and its set point was used as the output, to allow the NN to produce changing values of *kp* and *ki* for the controller, from one of the hidden layers. *Figure 5.8* displays the supply water temperature results for summer conditions as well as the controller gains. The average value for *ki* was 0.64 and the average value for *kp* was 1.53. It is important to note that the oscillations presented in the supply water temperature are acceptable in the single loop case. Since the models used in the project are quite stiff, when integrating this type of controller with the rest of the system tends to produce stability problems. In this case the set point was achieved by the controller in all the cases. Part of these better results might be due the necessity of the program to diminish the integration times in order to follow the changes given by the NN. The maximum error for the non-adaptive NN was -6.84C, its average error being -0.09C.

To continue the procedure, an adNN is developed to compare with the previous two cases. As a first attempt, the non-adaptive NN is converted into an adNN. The results depicted in *Figure 5.9*. The first thing that call the attention in this figure, is the lack of direct

visual improvement in the performance of the NN, however this is expected since the non-adaptive NN was properly trained for this particular system, and the conditions have not change dramatically. Nonetheless when comparing the values of *kp* and *ki* with those given by the non-adaptive NN, there is a noticeable decrease in the integral and proportional controller gains which leads to less oscillations in the response of the controller. This is reflected when evaluating the error of each of the systems. The maximum error encountered was -5.13C while the average error was 0.051C. In both cases the average error decreased in comparison with the non-adaptive NN (45%) and with the BC (92%). On the other hand, the maximum error was higher for the two types of NNs (almost double) compared to the base case scenario.

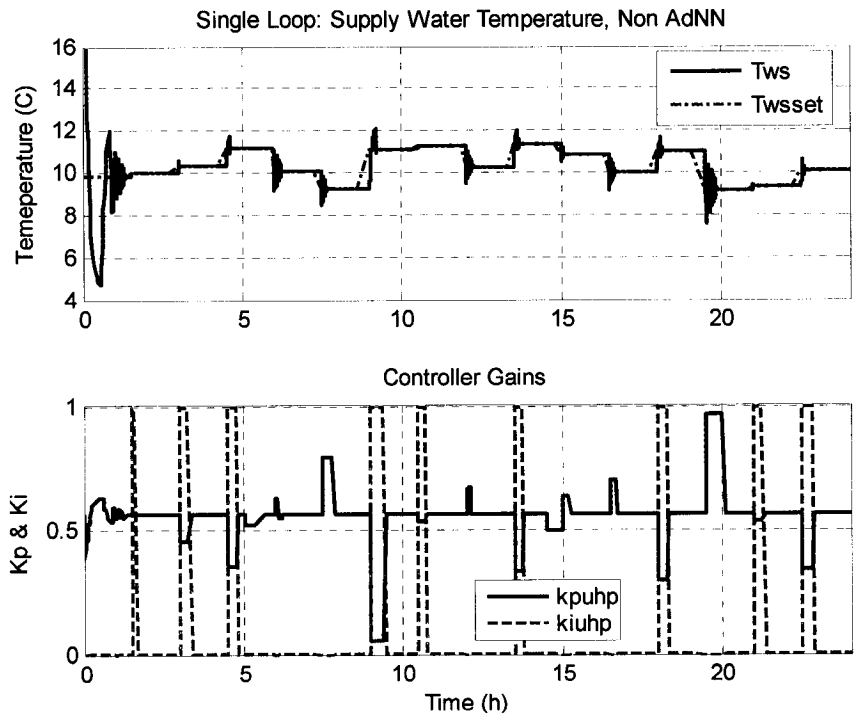


Figure 5.8. Single Loop Simulation for Non-Adaptive NN.

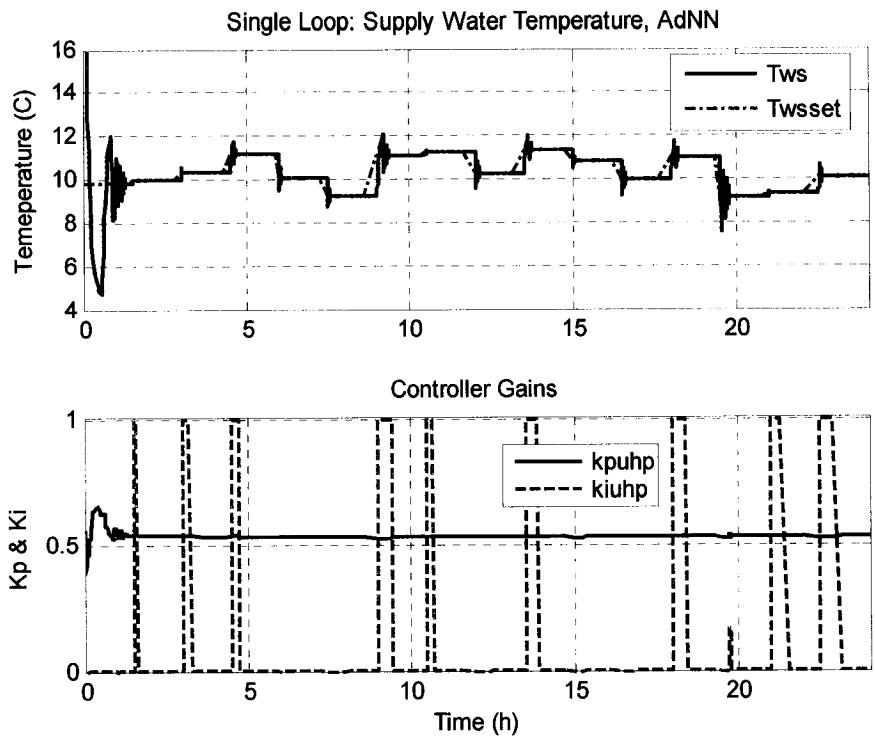


Figure 5.9. Single Loop Simulation for adNN.

This is due to the higher over and under shoot of the controller. Seeing no further advantage in the NN that was taken from the non-adaptive one, a new adNN with no previous training is developed. Also to diminish the time of calculations and to simplify the adNN architecture, the five step time delay for the same inputs is removed, as well as one of the hidden layers. The new NN architecture is [10, 2, 1], with transfer functions 'tansig', 'poslin', 'purelin'. *Figure 5.10* displays the results obtained for *Tws*. Again with a visual check, the benefits do not seem evident; nonetheless the NN performs to the same level as the pre-trained predecessors. Furthermore, the lack of tap delays ensures that the adNN can be applied at any point in the system, giving a faster response. Another advantage of this system is that it is not necessary to know in advance a set of controller constants with the purpose of pre-training the NN. Under these conditions the adNN showed an improvement in the under and over shoot, while maintaining the set point

tracking in all the cases. The maximum error encountered for the summer conditions was -5.13C, the average error being 0.014C. A comparison of the error of the supply water temperature with its respective set point between the three systems under different conditions is displayed in *Figure 5.11*. The maximum error for the constant *Kp* and *Ki* system was -2.88C while the average error was 0.68. The maximum error for the non-adaptive NN under these conditions was -0.13C while the average error was 0.11. The maximum error for the adNN was -5.14C while the average error was 0.09.

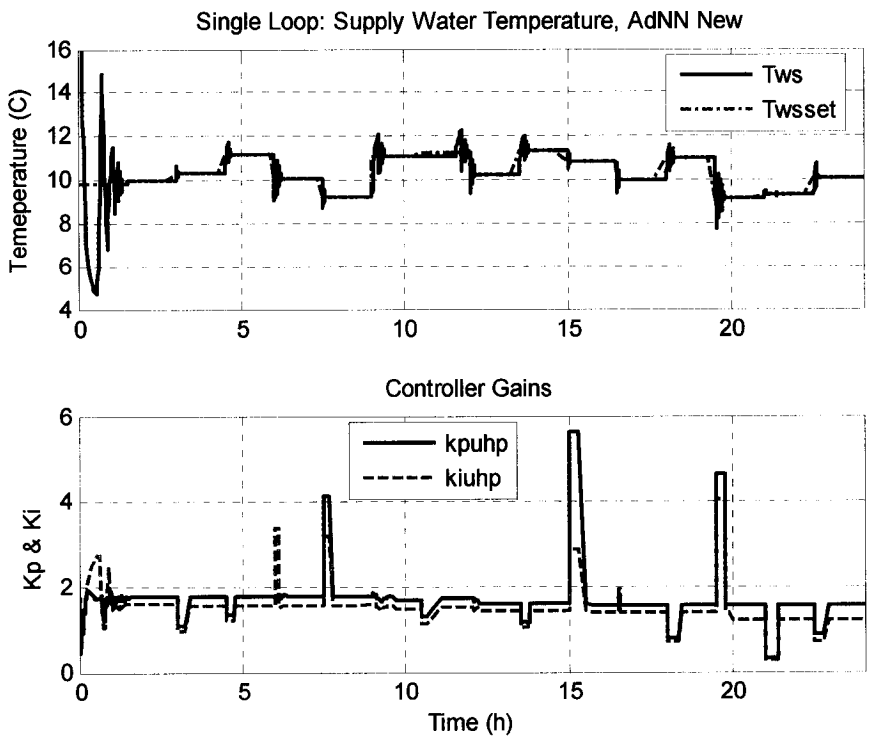


Figure 5.10. Single Loop Simulation for Adaptive NN New.

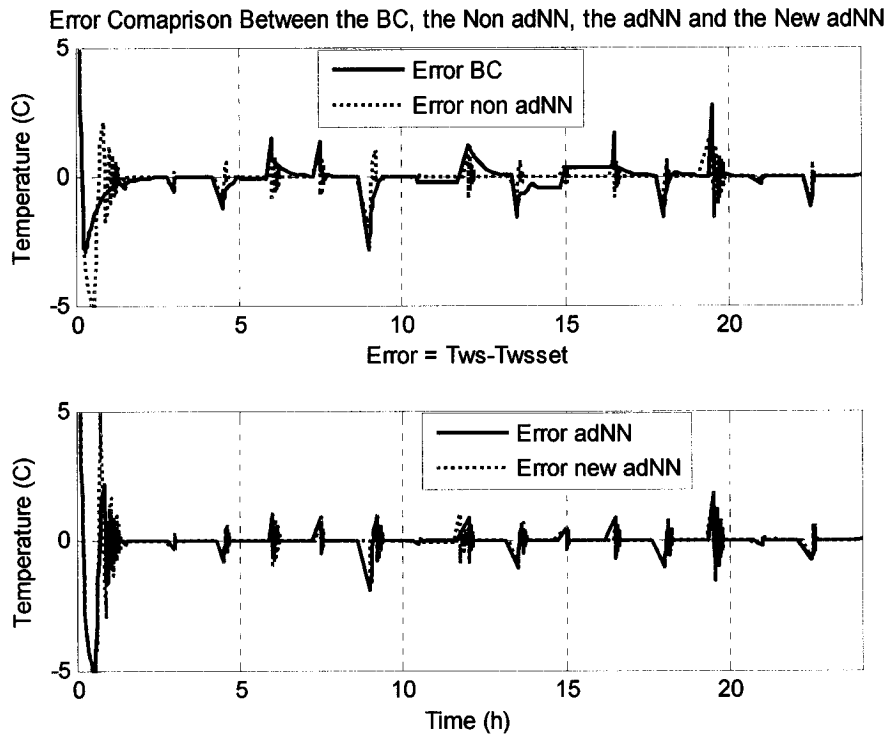


Figure 5.11. Single Loop Simulation for Supply Water Temperature, Error Comparison.

From the results is evident that the artificial NNs perform poorly compared to a well selected set of controller constants, however the adNN displayed an improvement in the performance during the second simulation run. It has a higher under and over shoot with some oscillatory behaviour, nonetheless the desired set point is achieved for most of the cases within 25 minutes of the set point change, compared to the base case, which takes around 40 minutes as it has a more damped performance.

5.3.3 Supply Air Temperature Loop

During the development of this particular adNN, a series of inconveniences were encountered. The system equations for this particular loop are very stiff, a fact that was evident even during the early stages of selecting the constant controller gains. The *kp* and *ki* used were very small, 0.009 and 0.0001 respectively, rendering some difficulties for

the NN to choose appropriate controller gains without reducing them to zero. As an approach to this problem, one of the differential equations of the model was converted in to a steady state equation, shown below.

$$\frac{dTt}{dt} = 0$$

$$Tt = \frac{\frac{Cpa \cdot Ma}{\rho \cdot A \cdot Lc}(Tsa - Ta,in) + \left(\frac{\eta so \cdot hc \cdot Ao}{\rho \cdot Cv \cdot A} + \frac{\eta so \cdot hc \cdot Ao}{mfin \cdot Cfin(1-\eta s)}\right)\bar{T}a + \frac{hit \cdot Ait}{mfin \cdot Cfin(1-\eta s)}\bar{T}w}{\left(\frac{\eta so \cdot hc \cdot Ao}{\rho \cdot Cv \cdot A} + \frac{\eta so \cdot hc \cdot Ao}{mfin \cdot Cfin(1-\eta s)}\right) + \frac{hit \cdot Ait}{mfin \cdot Cfin(1-\eta s)}} \tag{35}$$

Where *Tt* is the average temperature of the tube and *Tsa* is the air supply temperature. The rest of the variables are described in section 3.2.3. A comparison of the change in the system response with the two different average tube temperature equations (Equation (20) and Equation (35)) is made to corroborate if the change is feasible. *Figure 5.12* shows a 24h simulation run comparing the system response for the average tube temperature in the coil, under changing summer conditions. It also depicts the error comparison of this simulation run. *Figure 5.13* shows a close up of the simulation run for 10000 seconds (2h 45min). As it can be seen in the figures the differential equation (DE) tends to have greater over and under shoots than the algebraic equation (AE) during a drastic change in the supply air temperature set point. However when the changes are smaller the difference between the two equations are negligible. Given that the supply air temperature set point changes every 1200 seconds (20 minutes) according to the changes in the outside air temperature in the base case and every simulation step with the NNs, it is safe to assume that these changes will not be noticeable for the rest of the algorithm.

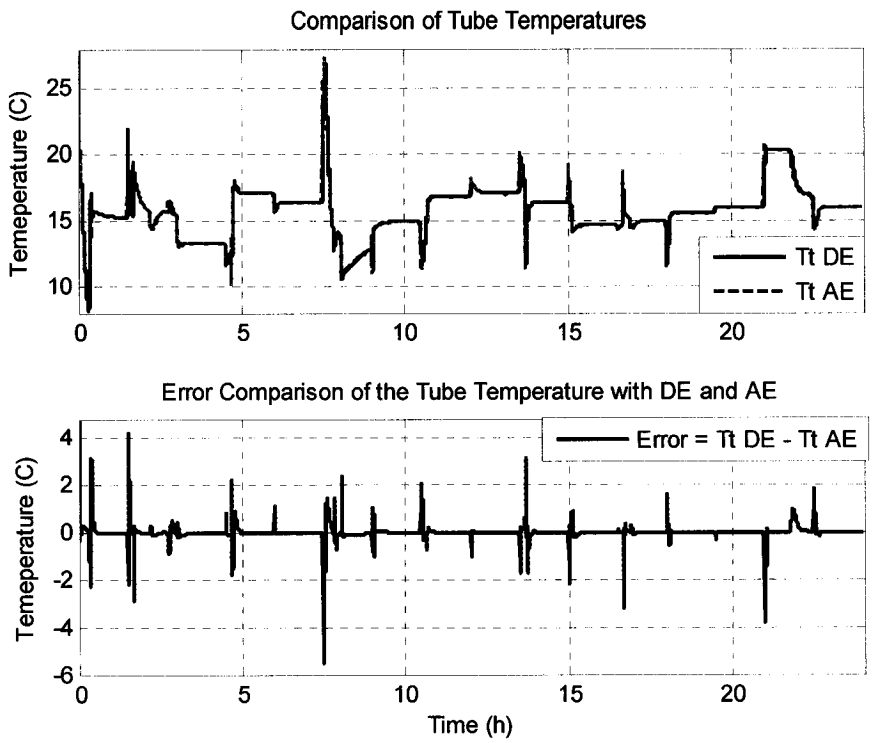


Figure 5.12. Comparison of the System Dynamics with a Differential Equation and the Algebraic Equation.

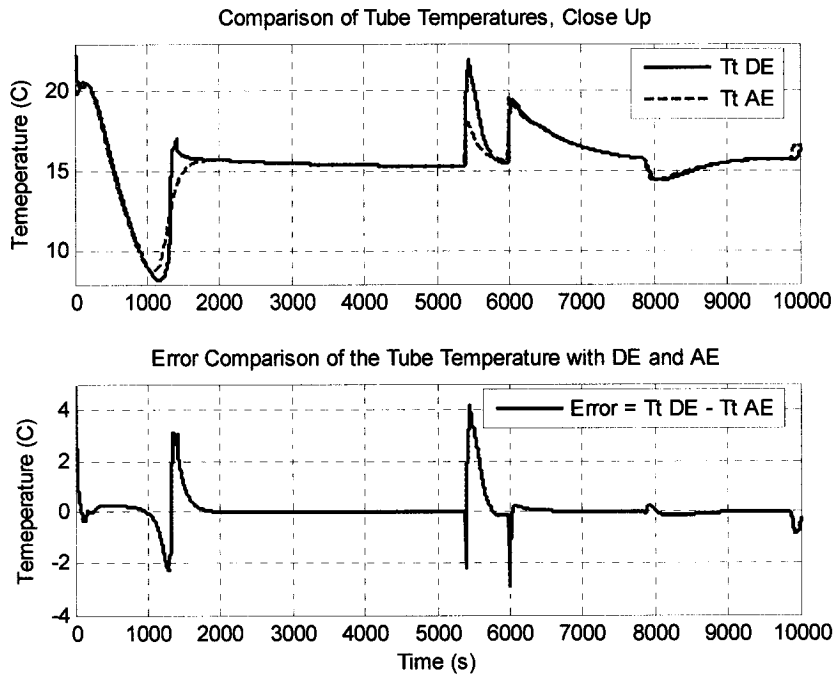


Figure 5.13. Close Up of Tt with a Differential Equation and the Algebraic Equation.

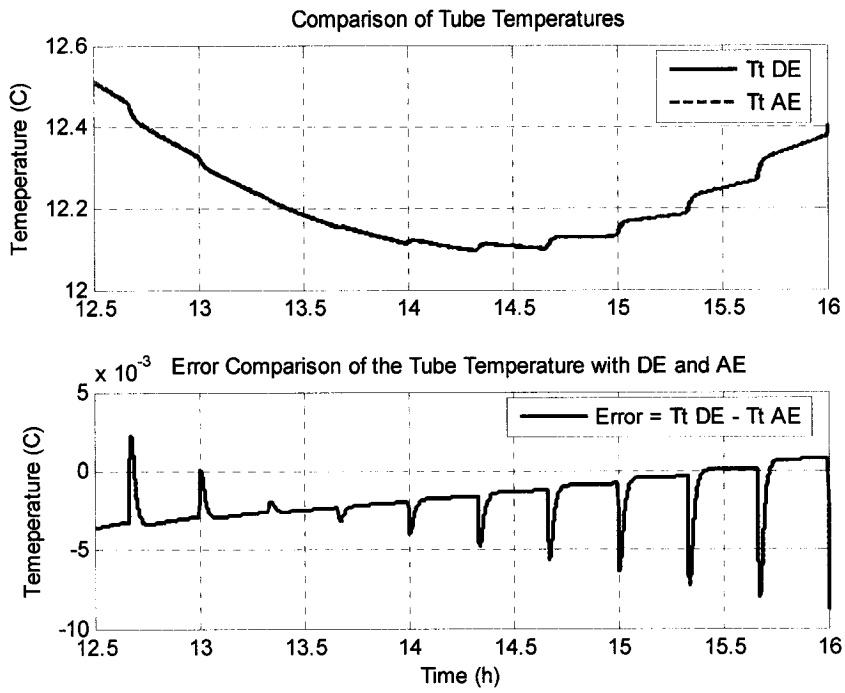


Figure 5.14. Error Comparison Between the Differential Equation and the Algebraic Equation in a 24h Simulation Run.

Proof of this is depicted in *Figure 5.14*. The figure shows a close up of the mean bulk tube temperature comparison of a 24h simulation run with the differential equation and with the algebraic equation, corroborating with the low error encountered that the difference between the two algorithms is insignificant, while the stiffness of the system is reduced allowing the controller gains to take higher values. Using this equation also has the advantage of reducing the computation time, allowing the adNNs to use this extra computational time for the associated calculations with the online training and data storage. By means of a system with the algebraic equation, the loop is simulated under winter and summer conditions. *Figure 5.15* shows in the top part the outdoor air temperature profile used for the three systems for the simulation runs while the lower part of the figure depicts the results obtained from the constant gain controller. *Figure 5.16*

displays the supply air temperature and its respective set point for the non adaptive NN while *Figure 5.17* shows the results for the adNN. Both figures display the changes in the controller gains in the lower part of them. *Figure 5.18* compares the error of the three systems.

For the non adaptive NN the same architecture used in section 4.2.5 is employed. A feed back [5, 2, 10, 1] NN with five tap delays for the inputs. The inputs include, the mass flow rate of water (*mw*), the supply air temperature (*Tsa*), the corresponding set point (*Tsaset*) and the supply water temperature (*Tws*), p = [*mw; Tsa; Tsaset; Tws*]. The NN was trained with the data obtained from the base case simulations. It is worth to note that the hidden layer that calculates the controller gains was selected to have a range between 0 and 1 to facilitate the process of selecting the values. For the adNN a similar architecture was used: a feed forward [5, 2, 6, 1] with the same inputs, but without the tap delays. This reduction in the number of neurons proves beneficial for the reduction in computational times for the online training process. This adNN is not pre-trained for after being created it is just used in the system and trained online to supply an appropriate set of controller gains. In this case the hidden layer that procures the controller gains limits its values between 0 and 100. From the figures it is evident that both NNs perform to the level of the selected *kp* and *ki* constants. In most of the cases the NNs reached the desired set point in less than a quarter of the time than the case with the constant controller gains. Still the shorter time for reaching the set points is penalized with higher over and under shoot values at each change of set point.

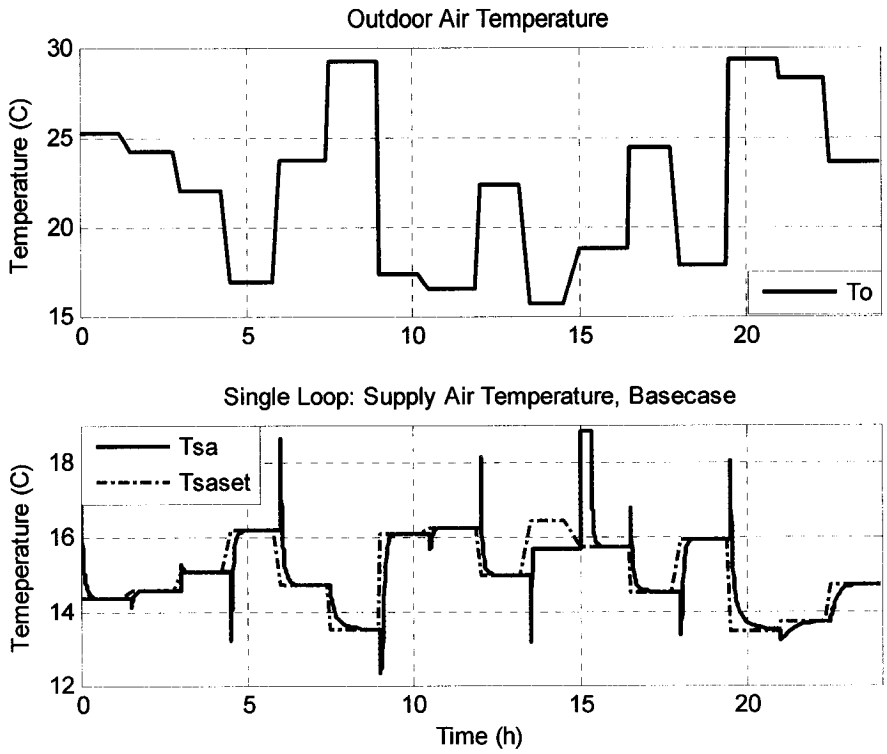


Figure 5.15.TSA Loop, Constant Kp and Ki.

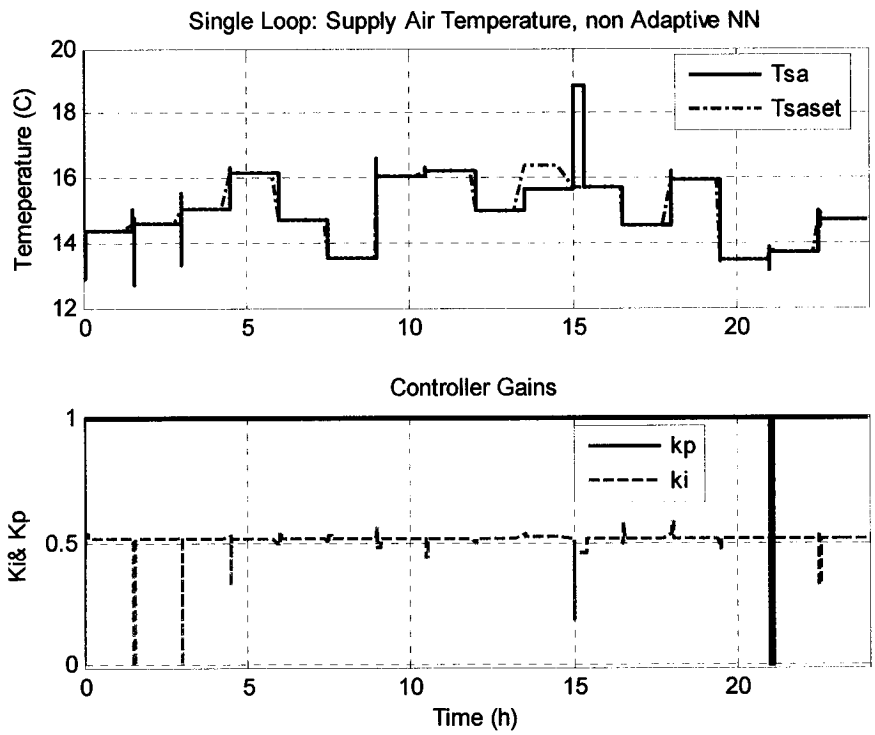


Figure 5.16.TSA Loop, Non-Adaptive NN with Variable Kp and Ki.

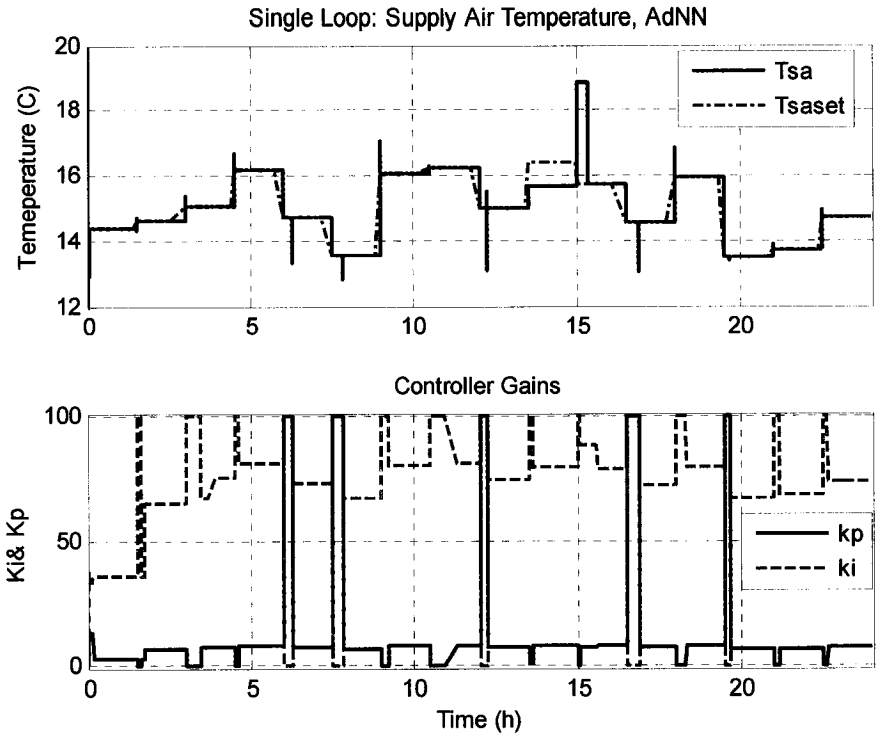


Figure 5.17.TSA Loop, Adaptive NN with Variable Kp and Ki.

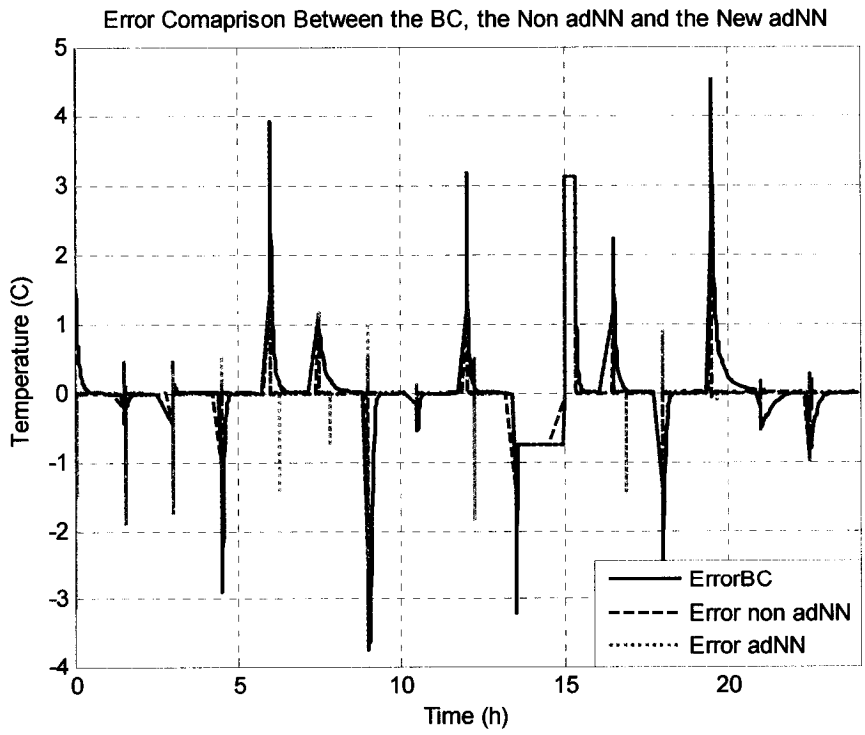


Figure 5.18.TSA Loop, Error Comparison.

It is worth to note that in all three cases there is a period of time in which neither of the algorithms reached the required set point. This error is due to the selected integration times in the program. In the period of time between 13.5h and 15h the algorithm reduces the mass flow rate of water to 0 to force the supply air temperature to rise. Even so the set point is not reached, and at the next change in the conditions the program takes full 30 minutes of the simulated time to change the requirements of mass flow rate of water to satisfy the new conditions. In this period of time the temperature rises giving the high error displayed in all the figures. The maximum error displayed in *Figure 5.18* for the BC algorithm was 4.51C while the average error was -0.67C. For the non adaptive NN the maximum error was 3.11C, produced by the integration time error. The average error for this case was 0.13C, considerably lower than that of the BC. The maximum error for the adNN was again 3.11C, once more due to the integration time selected by the program. The average error was 0.12C. The NN performed altogether better than the constant controller gain system. However the difference between the non-adaptive and the adaptive NN is not in performance, but in the easiness of implementation. The adNN lacking the need for initial training data would be more adequate to generalize the need of a real system.

5.3.4 Zone Temperature Loops

The zone temperature control loop was considered as a single case, since the two zones are very similar in their dynamic behaviour. In order to compare the different results obtained from the BC scenario, with constant *kp* and *ki*, from the non-adaptive NN with variable controller gains and from the adNN also with variable *kp* and *ki* two cases were considered. First the response of the three systems under summer conditions is presented.

For the simulation runs the supply air temperature is fixed to the set point and provided by equation (24a) (see Chapter three).

$$Tsaset = -0.2 \cdot To + 17.8746 \qquad (24a)$$

The outdoor air temperature is changed randomly according to the desired season to simulate. A similar procedure was employed to imitate the cooling loads in the zones. In *Figure 5.19* the outdoor air temperature is depicted, as well as the supply air temperature selected by equation (24a). It is also shown in the same figure the results of the simulation run under constant *kp* and *ki* for the controller. *Figure 5.20* shows the results for the non adaptive NN with variable *kp* and *ki*. The lower part of the figure shows the values for the controller gains determined by the NN during the simulation run. *Figure 5.21* depicts the results for the adNN with variable controller gains while *Figure 5.22* shows a comparison of the difference between the zone temperatures achieved by each one of the systems with the respective set point. The non-adaptive NN architecture that showed good results was a three inputs (*Tz*, *Tsa*, *ma*) with five delays,[5,2,1], transfer functions 'tansig', 'poslin', 'purelin'. A simplified version of this NN was used for the adaptive purposes, also three inputs (*Tz*, *Tsa*, *To*), [3, 2, 1] with the same transfer functions and no delays displayed good results. From *Figure 5.22* is noticeable that the three systems reach the desired set point; however, the non-adaptive and the adaptive NNs present some oscillation before reaching it. This is due to the change in the set point. Nonetheless this also proved that both NNs perform to the level of the selected controller gains. The change in the inputs was to improve the predictor capacity, allowing the adNN to evolve according to the data presented from the system. It is worth to note that the non adaptive NN was trained offline with data obtained from the BC, while the adNN was

simply connected to the system and adapted to its requirements. Also it was proved that the non adaptive NN will require new training when a drastic change in the conditions in the system occurs, like from summer to winter. For the adNN, since the objective was to find if it was able to cope with the system without previous training a new adNN was used with the simulated winter conditions. Again the results proved to be adequate. Nonetheless an improvement in the time response would be good, to avoid the actuators from damage for over positioning. The maximum error for the BC (constant *ki* and *kp*) was -3.08C while the average error was 0.026. For the non-adaptive NN with variable *kp* and *ki*, the maximum error was the same as for the BC -3.08C, but the average error was slightly higher 0.028. The maximum error for the adNN with variable *kp* and *ki* was again the same (-3.08C), however the average error was 0.005, almost half of the other two values, implying that the set point was reached faster than in the other two cases.

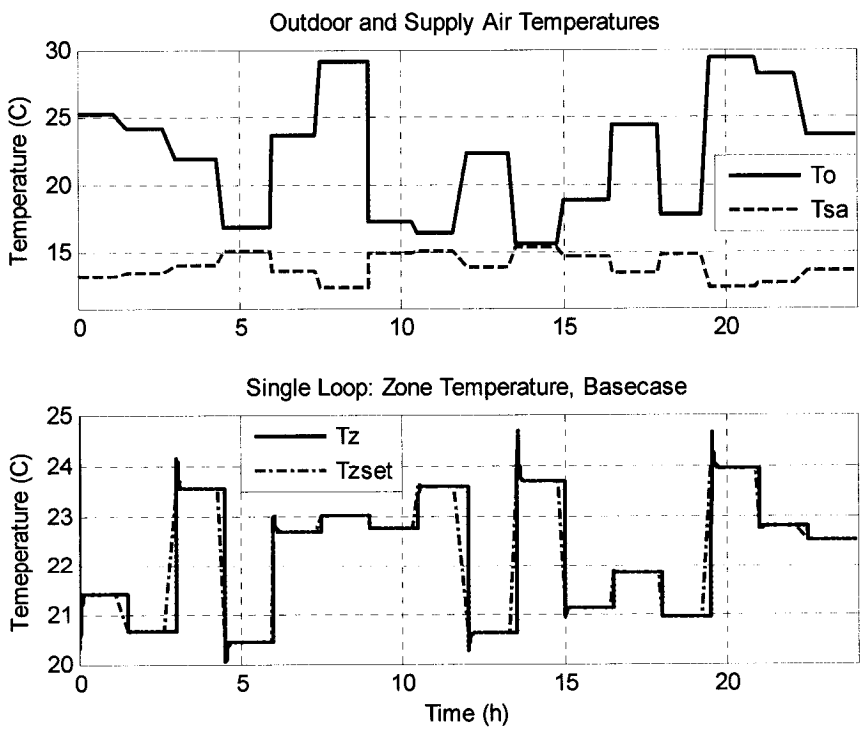


Figure 5.19. Single Loop Simulation for Fixed Kp and Ki.

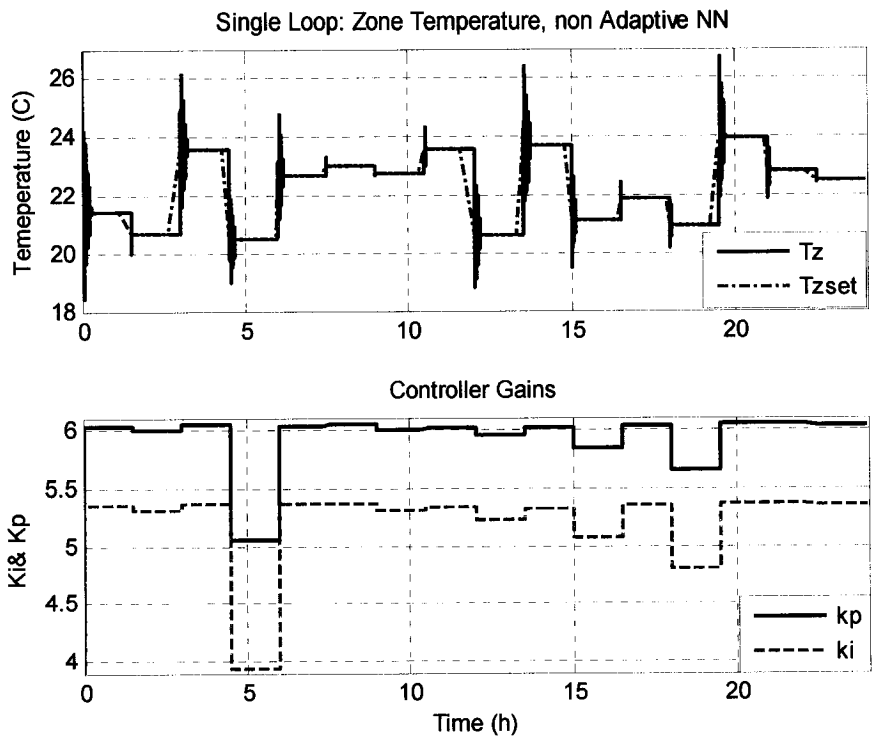


Figure 5.20. Single Loop Simulation for Non-Adaptive NN.

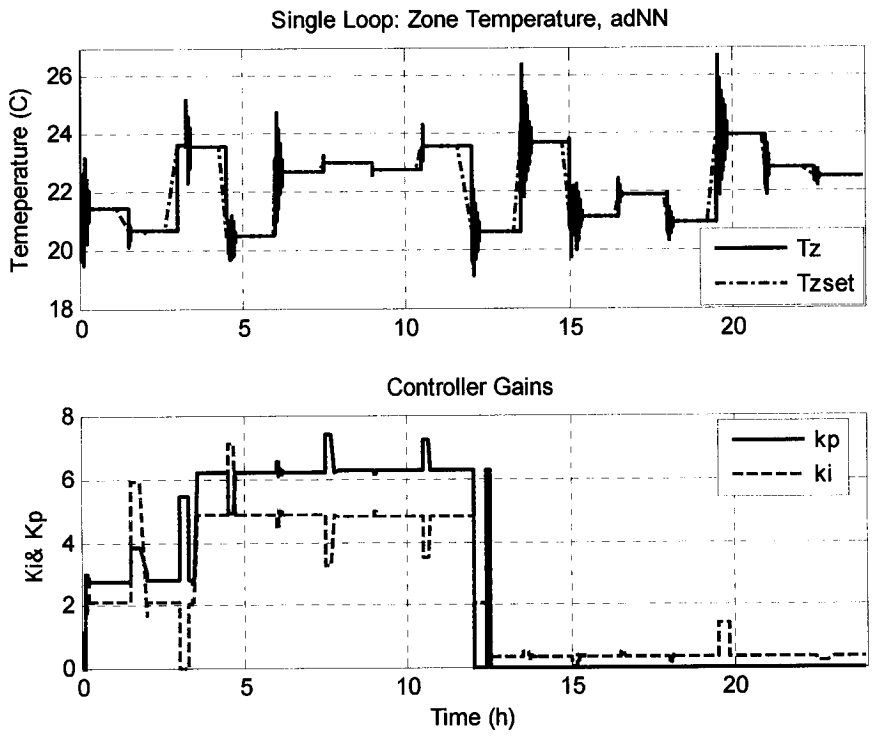


Figure 5.21. Single Loop Simulation for Adaptive NN.

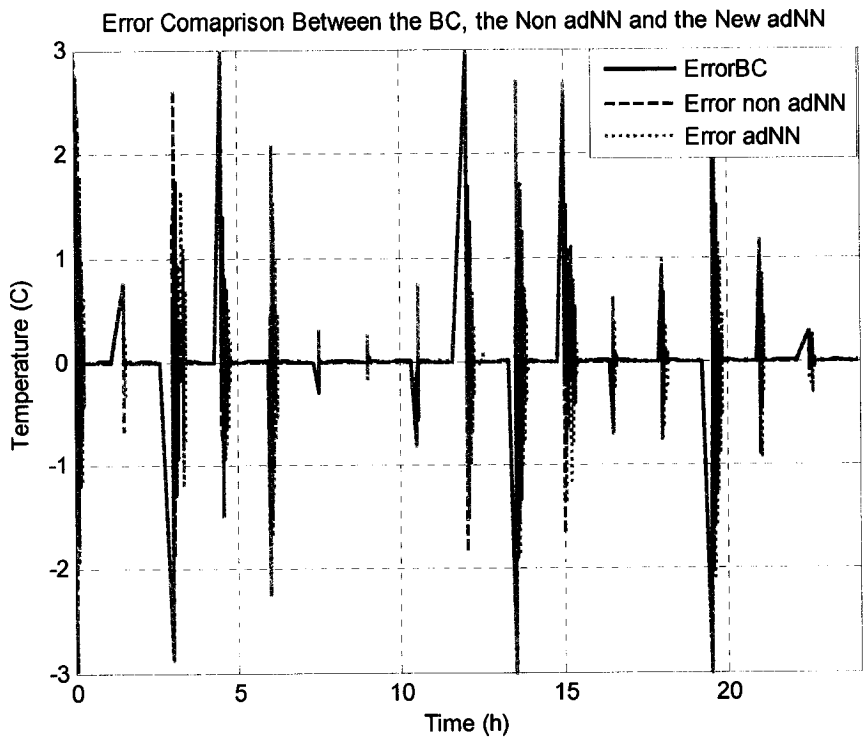


Figure 5.22. Zone Temperature Error Comparison.

The improvement in the overall performance of the adNN is not evident; however, it has the advantage over the non adaptive system of not requiring any previous data to train and possibly avoiding the need of manually selecting the controller gains while still reaching the desired set points.

5.3.5 Integrated PI – Neuro – Controllers

After developing the adNNs for each one of the control loops, it is necessary to integrate them in order to see the combined effect in the system. *Figure 5.23* shows the supply water temperature compared with the desired set point, *Figure 5.24* displays the supply air temperature, while *Figure 5.25* and *Figure 5.26* depicts the zone 1 and zone 2 temperatures. For each figure the value given by the hidden layer of the adNN to the controller gain, this is depicted in the lower part of them. It is clear that the adNN are able

to still produce good results when the dynamics of the system are changed. Once again, it is worth to note that these adNNs had no previous training and consequently no need for a training data set. Referring to the behaviour of the controller gains, it is evident that the NNs tend to keep the values of the integer controller gain constant, with some exceptions in the zone temperature graphics, and the main change is generated by means of the proportional gain, as in Chapter 4. All the figures were generated by the system with the simplified dynamics explained in the section 5.3.3, where the differential equation for the mean bulk tube temperature was replaced by the algebraic equation due to the small range of values accepted as controller gains. A comparison between the errors of the three integrated systems is made in *Figure 5.27* and *Figure 5.28*.

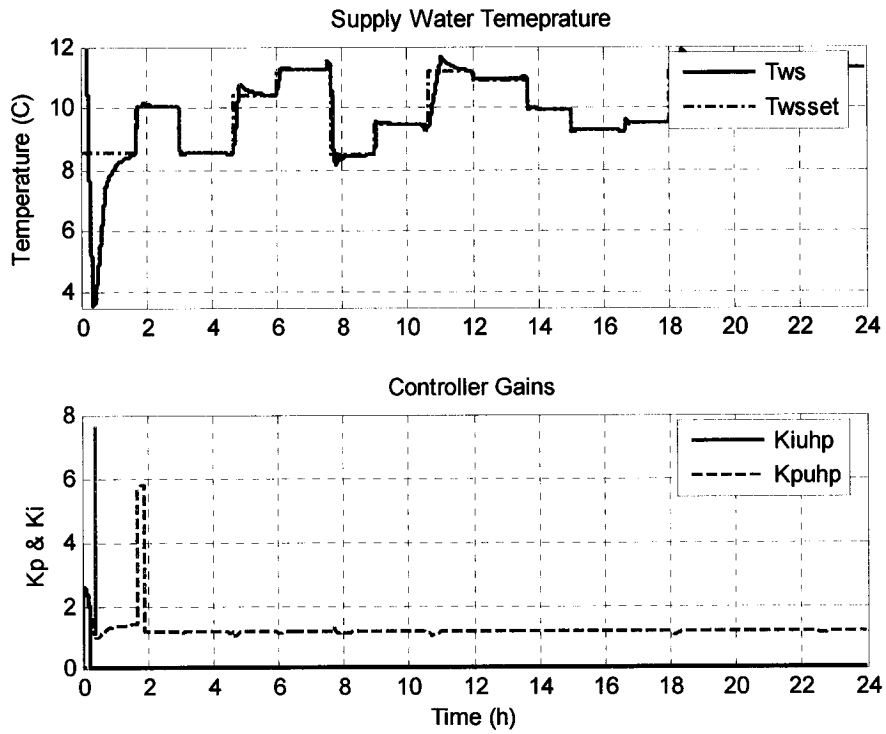


Figure 5.23.Supply Water Temperature Adaptive NN.

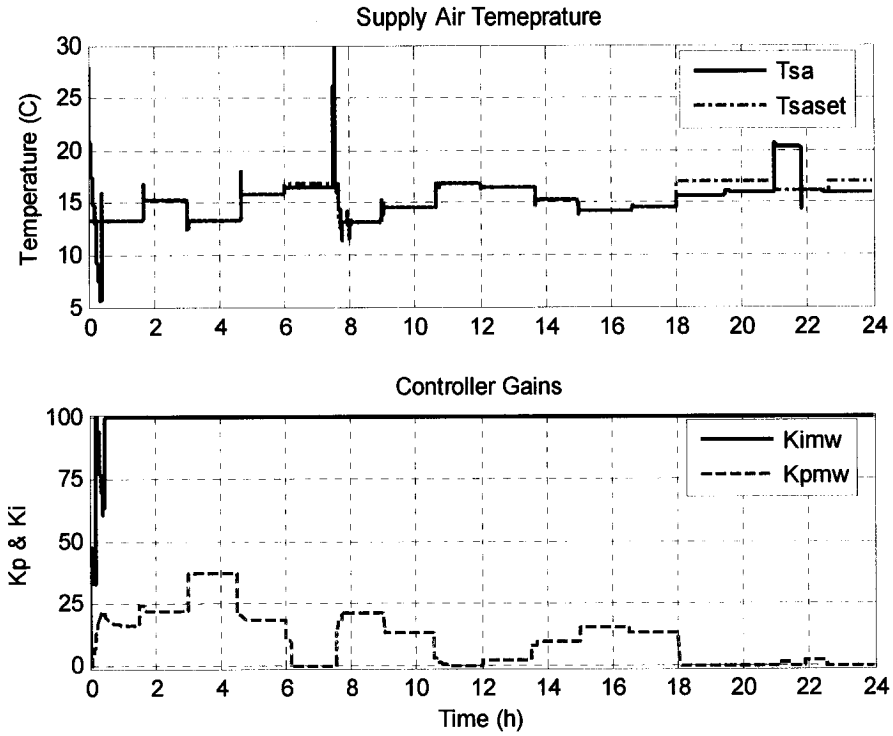


Figure 5.24. Supply Air Temperature Adaptive NN.

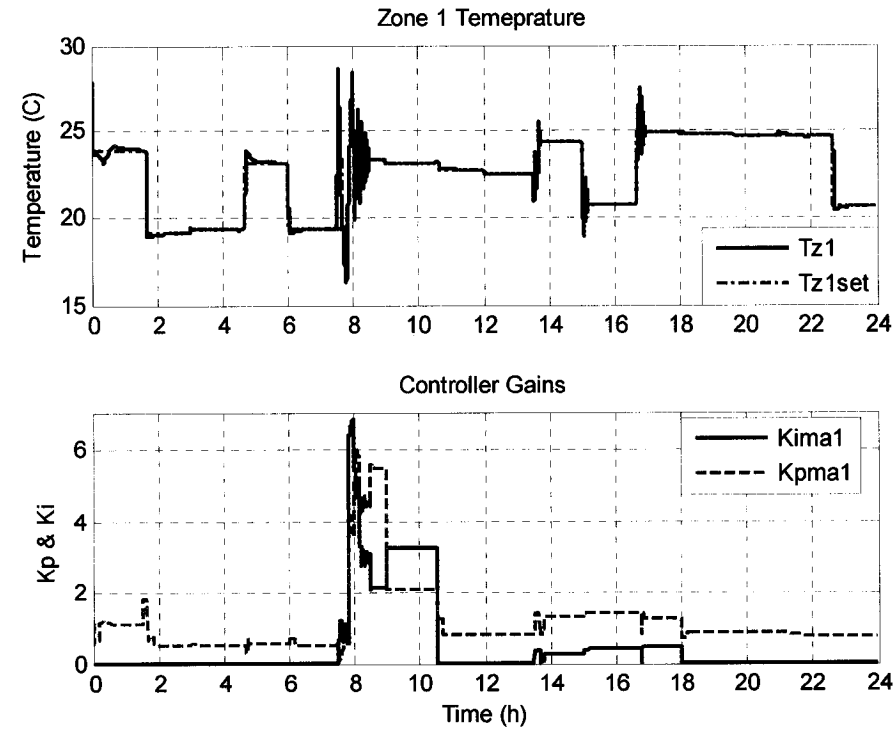


Figure 5.25. Zone1 Temperature Adaptive NN.

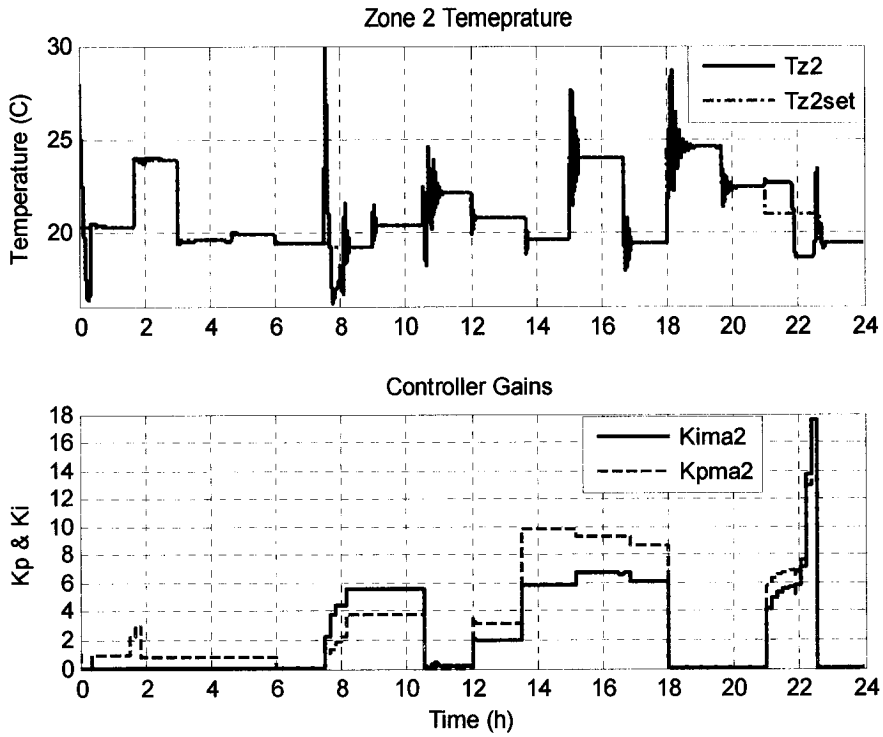


Figure 5.26. Zone2 Temperature Adaptive NN.

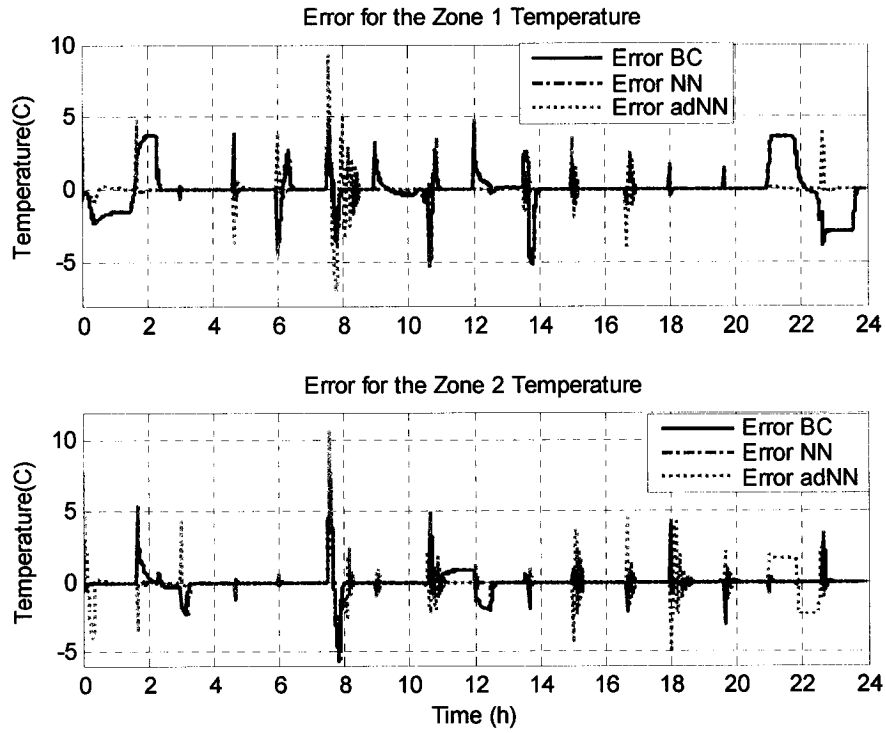


Figure 5.27. Zone Temperature Error Comparisons.

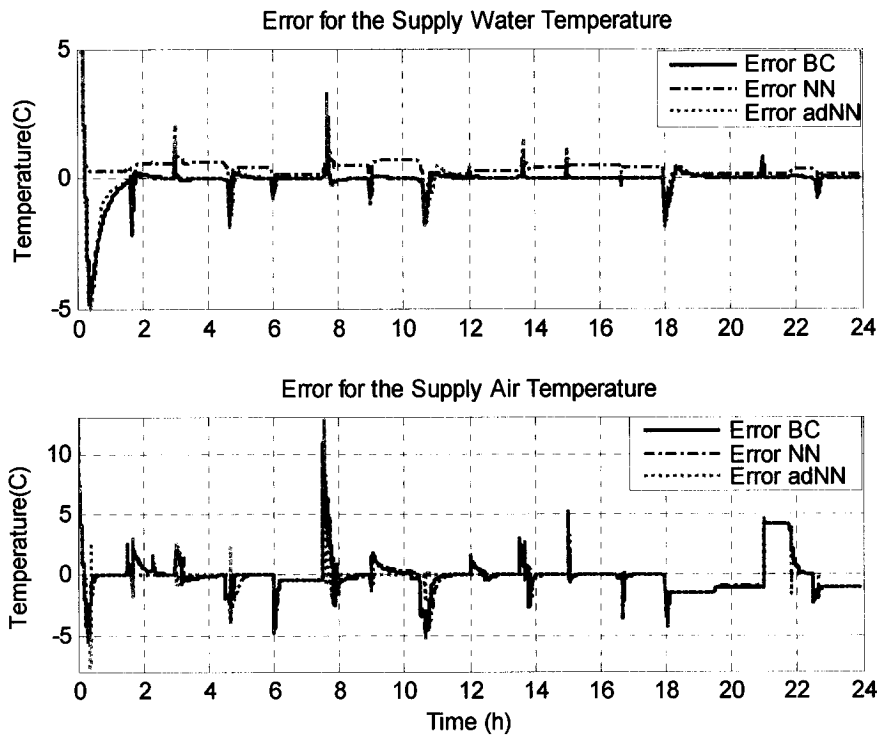


Figure 5.28. Supply Air and Water Temperature Error Comparisons.

During the coupling of the adNNs, certain difficulties were encountered. Since there is no previous training involved, if one of the adNNs diverges from the desired behaviour, then the whole system will diverge. This is expected due to the highly interactive nature of the system, being the output of one adNN the input of the next and vice versa. This matter can be solved in different ways. One of them can be to reinitialize the adNNs each time the results are not satisfactory. This solution is highly viable in a simulation, but it might be more complicated in a real system. Another possibility is to initialize the adNNs one by one, giving the chance to the system to stabilize before the introduction of a new adNN. Then again this possibility takes away the advantage that the adNNs did not require previous training or knowledge of a working set of controller gains to be implemented in the system. Here a single adNN is presented as a hypothetical alternative. This adNN receives as an input the outdoor air temperature (*To*), the supply water

temperature (*Tws*), the supply air temperature (*Tsa*), and the zone temperatures (*Tz1* and *Tz2*), and has as an output the difference between the set points of the four control loops (the two zones and the supply air and water temperatures); From the hidden layer the controller gains for the four loops are obtained while keeping a simple three layer architecture [10, 8, 4], with transfer functions 'purelin', 'poslin', and 'purelin'. It is worth to note that the hidden layer output has a limited range between 0.001 and 100.The *Figure 5.29* and *Figure 5.30* displays the zone temperatures and the air and water supply temperatures while *Figure 5.31* and *Figure 5.32* show the control variables and controller gains provided by this single NN .The errors for all control loops are depicted in *Figure 5.33*.

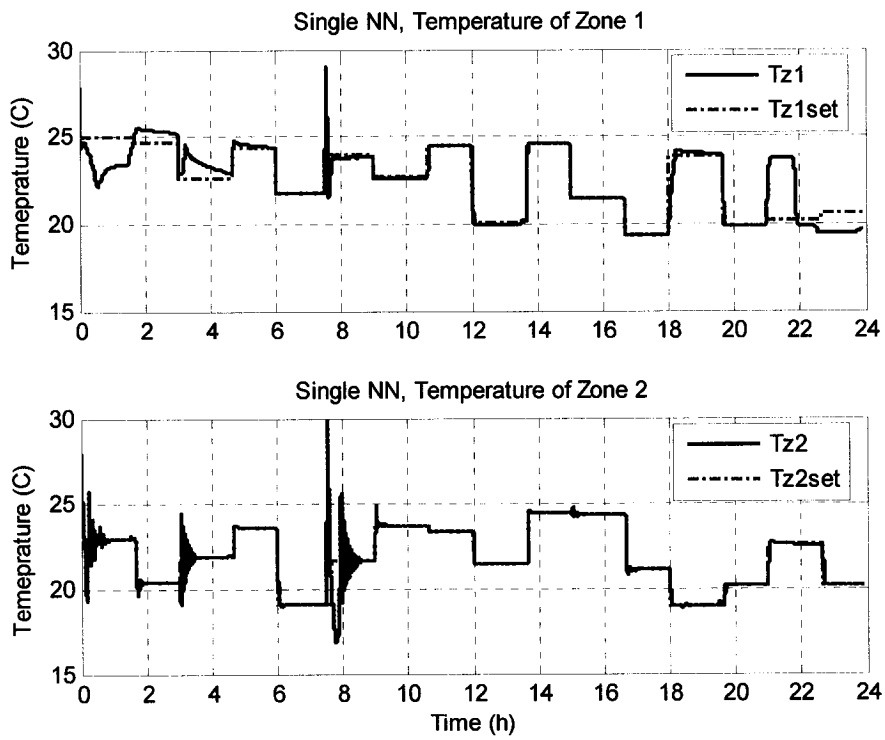


Figure 5.29. Zone Temperatures, Single Adaptive NN.

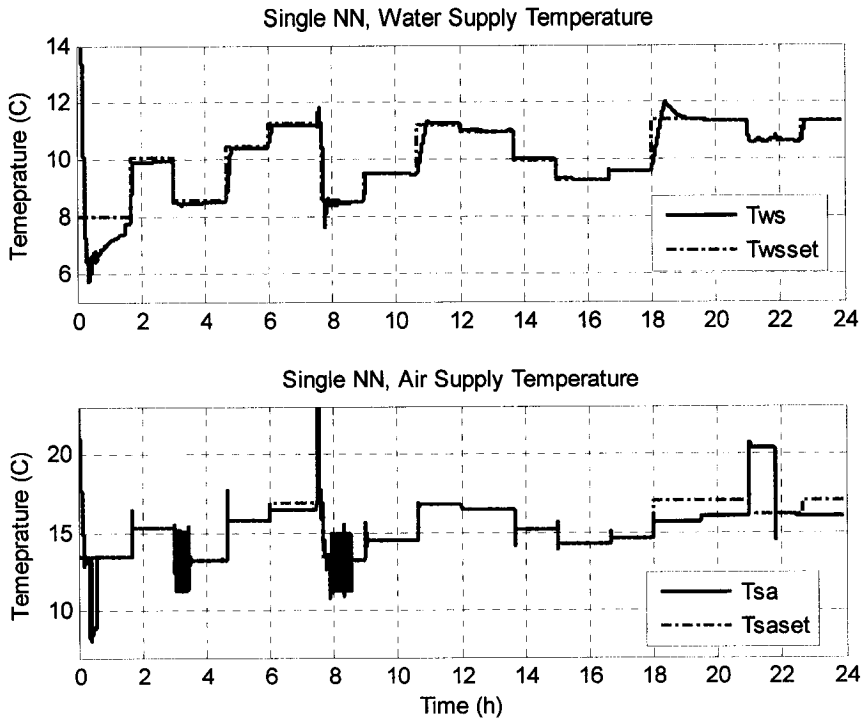


Figure 5.30.Supply Water and Air Temperatures, Single Adaptive NN.

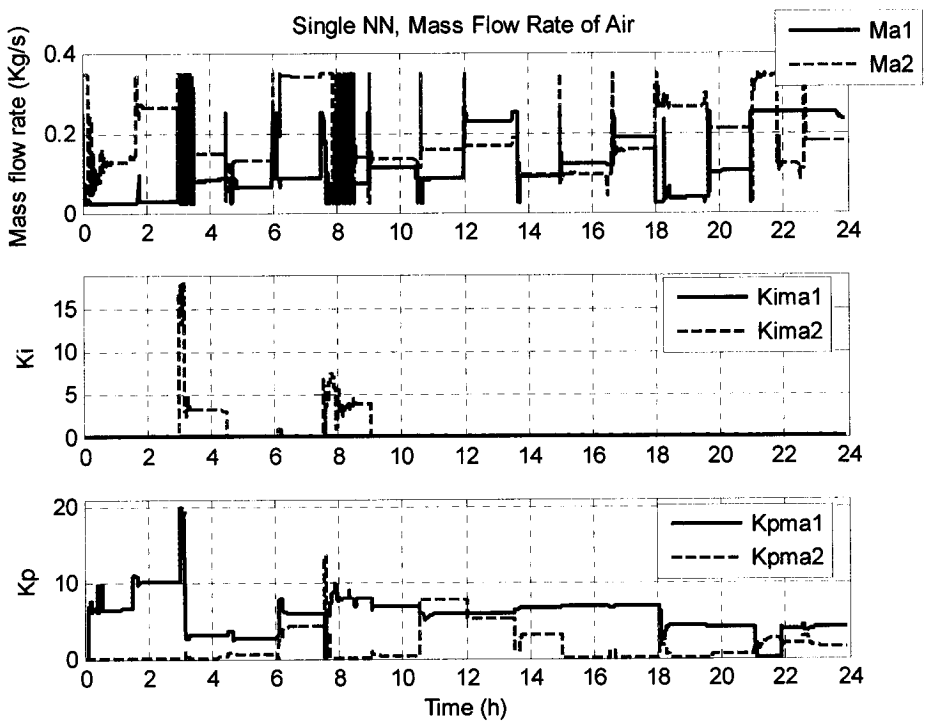


Figure 5.31. Zone's Mass Flow Rates of Air and Controller Gains, Single Adaptive NN.

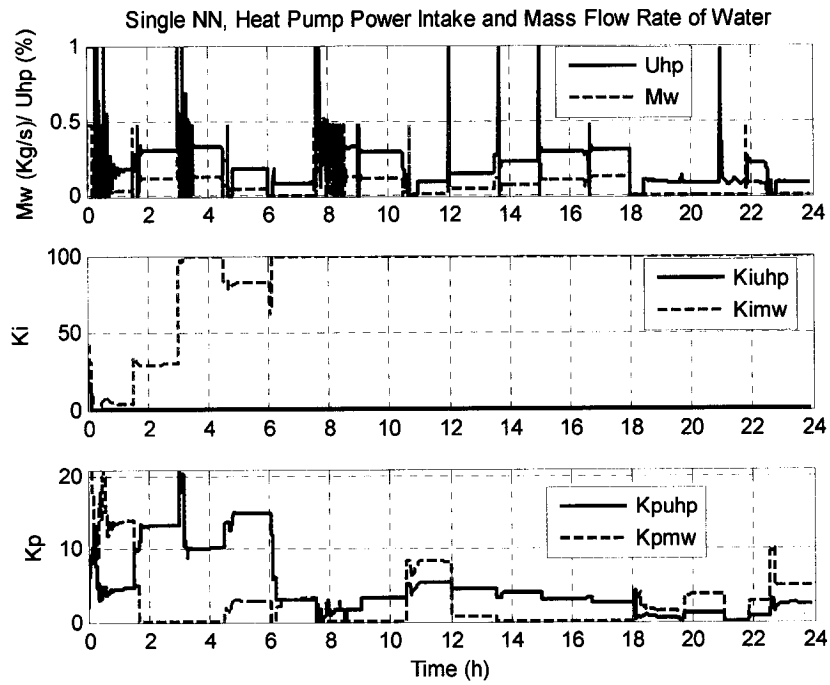


Figure 5.32. Uhp, mw, and Controller Gains, Single Adaptive NN.

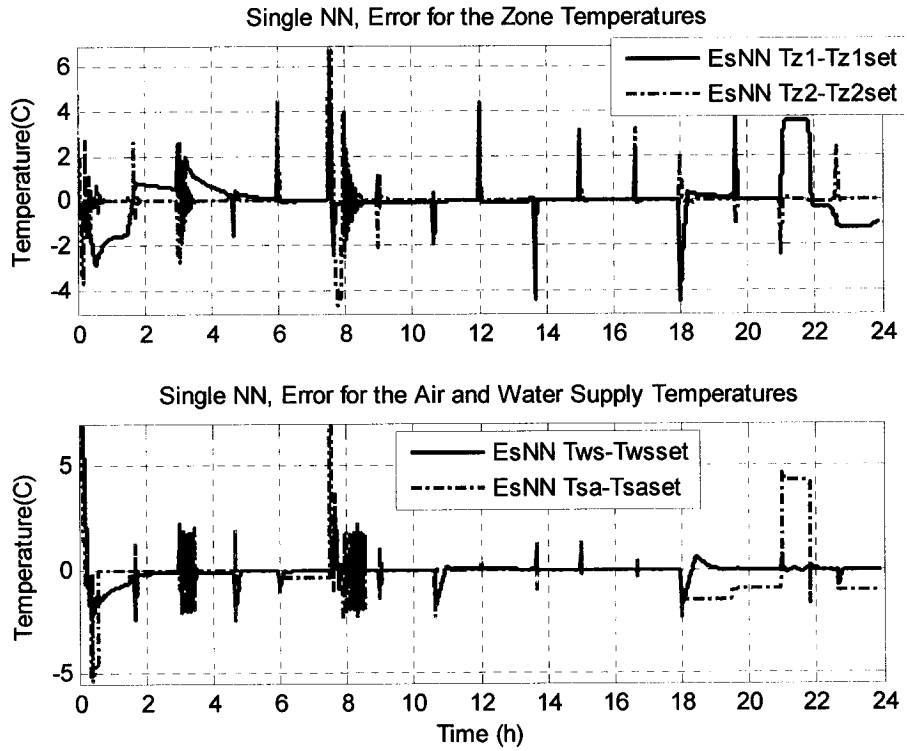


Figure 5.33. Errors of the Tz1, Tz2, Tws and of the Tsa , Single Adaptive NN.

From the figures it is clear that the system could be manage by a single NN providing the gains for the controllers. The single NN achieves the set point in 95% of the cases for the supply water temperature, in 70% of the cases in the supply air temperature, in 75% of the cases for the zone 1 and in all cases for the zone 2.

Loops (C)	**Tz1**		**Tz2**		**Tws**		**Tsa**	
Strategy	Max	Ave	Max	Ave	Max	Ave	Max	Ave
Base Case	5.9	0.07	9.2	0.00	2.02	0.02	8.53	-0.03
NN	6.8	0.07	10.2	0.00	3.43	0.44	8.96	-0.03
Ad NN	9.3	0.01	10.8	0.01	2.81	0.01	14.71	-0.10
Single Ad NN	7.3	0.05	11.0	0.01	2.31	-0.02	14.50	-0.05

Table 5-4. Error Comparison of the Different Strategies for PI Controller Gain Selection

Table 5-4 summarizes the maximum error and the average error for each one of the control loops, according to the strategy used.

It is evident from the results that all NNs have higher over shoots than the selected PI controllers, however the performance of the adNN is better than the non adaptive ones in the sense that the set point are reached with smaller steady state error. The difference between the adNNs working separately and the single adNN is small; nonetheless it proved to be easier to apply a single NN than to combine four different ones. Perhaps, it would be reasonable to assume than in a real system would be also more convenient.

5.4 Integrated Adaptive Neural – EMC System

The purpose of this section is to combine the generated adNNs in a integrated adaptive Neural EMC system. Three cases are compared in this section: the BC, the non adaptive NNs and the adNNs hybrid system. For the adNNs hybrid system the PI gain controller selector the single adNN was used due to its similar performance to the individual adNNs yet using the advantage of the easiness in implementation.

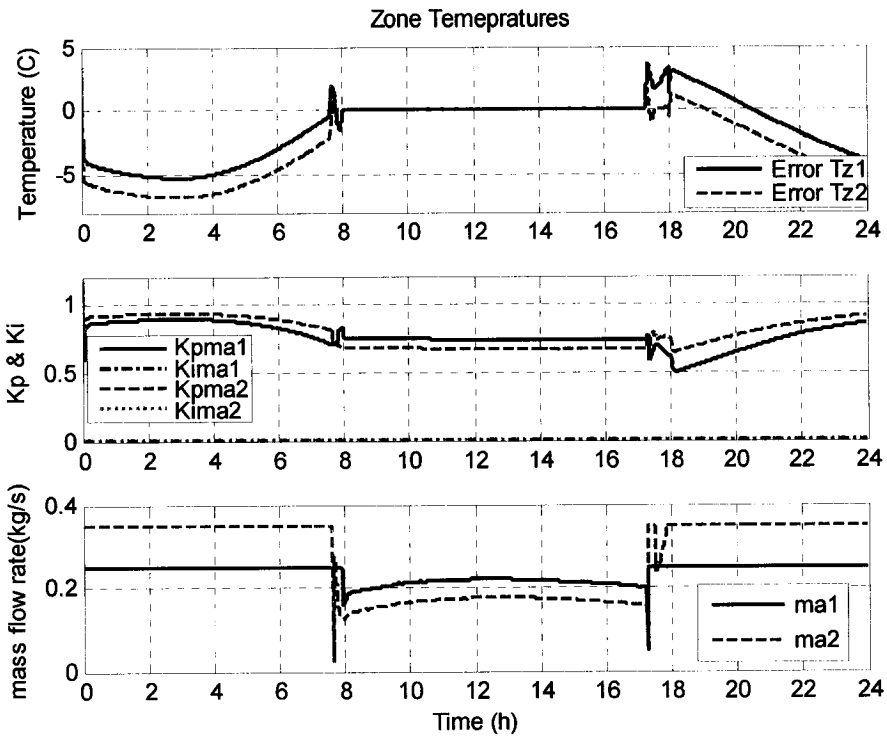


Figure 5.34. Errors for the Tz1 and Tz2, Controller Gains and Mass Flow Rate of Air.

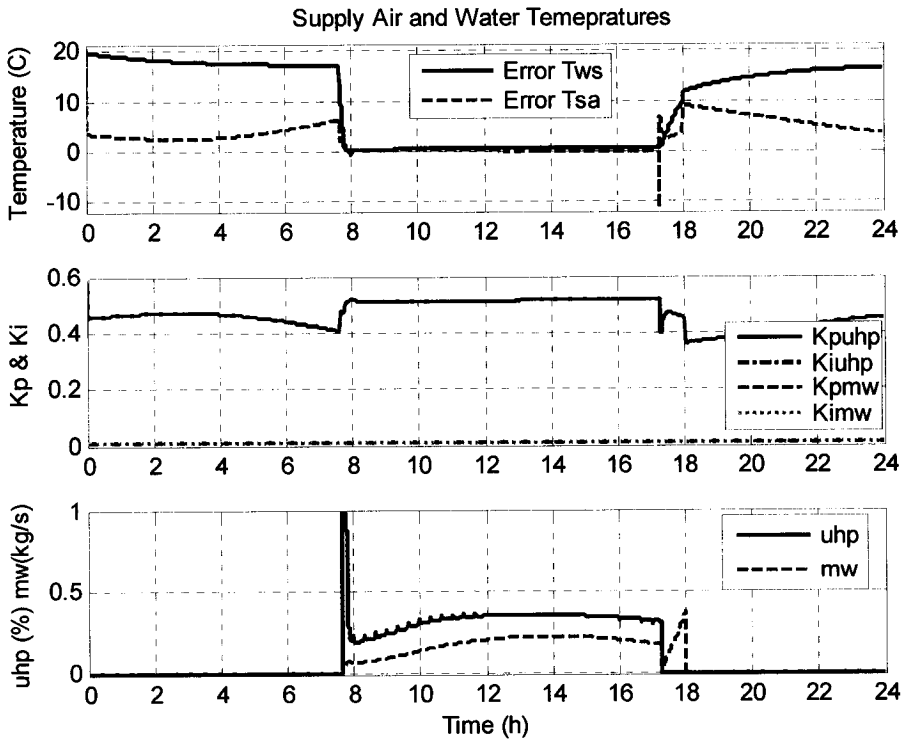


Figure 5.35. Errors for the Tws and Tsa, Controller Gains and mw and uhp.

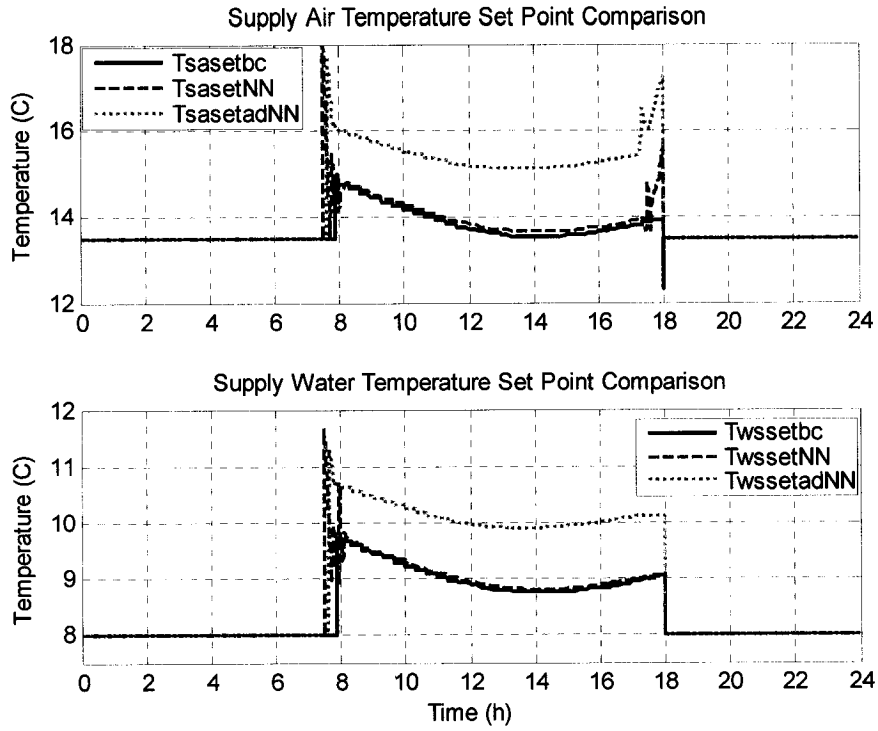


Figure 5.36.Supply Air and Water Set Point Reset Comparison.

Also the outdoor air intake non adaptive NN was used, since no adNN was developed. *Figure 5.34* shows the error for the zone temperatures, the controller gains and the mass flow rate of air. *Figure 5.35* displays the errors for the supply air and water temperatures, the correspondent controller gains and the mass flow rate of water and power input. *Figure 5.36* displays the comparison of the supply air and water set points given by the three cases.

Strategy	**BC**		**NN**		**AdNN**	
Loops (C)	Max	Ave	Max	Ave	Max	Ave
Tz1-Tz1set	-5.29	-0.001	-5.51	-0.001	-5.31	0.003
Tz2-Tz2set	8.39	0.000	8.49	0.000	3.65	0.002
Tws-Twsset	-3.49	0.016	-6.09	0.040	0.33	0.445
Tsa-Tsaset	-9.5	0.274	-10.9	0.085	-11.4	0.155
Start Time	28324 (s)		26977 (s)		27499(s)	
Stop Time	64799 (s)		62885 (s)		62221(s)	
Energy	219.1418 (MJ)		215.7537 (MJ)		201.5952 (MJ)	

Table 5-5. Error Comparison of the Different Strategies

Table 5-5 displays the error comparison between the three systems. While *Figure 5.37* shows the total and hourly energy consumption comparison. From these results we can conclude that the system performs as well as the BC algorithms, the tracking performance was improved from that of the non adNNs and with the advantage of requiring hardly any previous knowledge of the system, while still performing under the ongoing dynamic changes in it. However there were no major improvements in the energy consumption from the BC algorithm. It is suggested to create a series of predictors to feed and optimizer to provide better results. Yet the computational cost of such a system will render difficulties in implementing it in a real system.

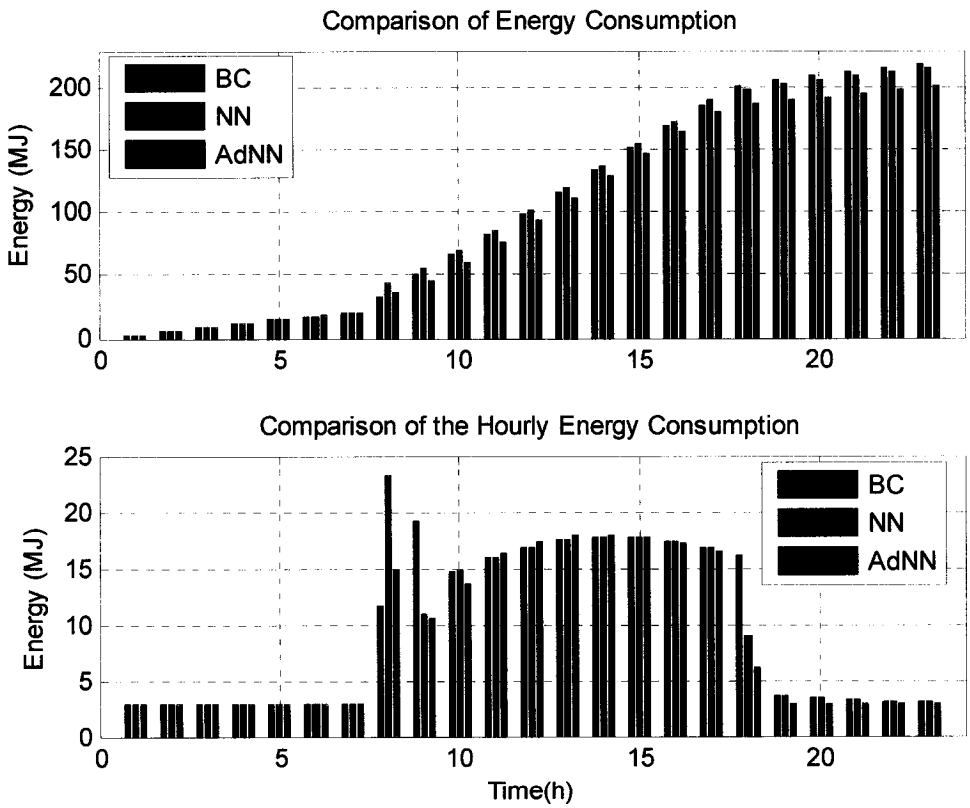


Figure 5.37. Energy Consumption Comparison.

5.5 Conclusions

1. The adNNs are able to compete with the results obtained by both the BC and the non adaptive NNs. The tracking performance and the overall energy consumption were improved; however no drastic enhancement was achieved.
2. There are two main advantages to the adaptive Neuro EMC system. The first is the ability to cope with ongoing changes in the system, which includes seasonal changes and in equipment. The second is related to the ease of implementation. With hardly any previous knowledge of a system, the adNNs are able to learn and generalize the dynamics of the system, providing good controller gains and EMS with energy savings comparable to those given by the conventional algorithms.
3. The start and stop lead time NNs, once trained online, display good prediction in the lead time values without compromising the comfort of the occupants. The resultant savings are not evident due to the late start of the BC algorithm. Nonetheless, the overall performance of these adNNs presents a significant improvement from that of the non adaptive NNs. It was also noticeable that the performance of the adNNs improved with the number of simulation runs and data acquired. Thus the general performance of the system would also be expected to improve with time.
4. The adaptive PI- Neuro controllers that were developed improved the tracking performance. It was clear that the integral controller gain generated by the individual adNNs was maintained almost constant throughout all of the simulation runs, similar to results obtained in the non adaptive cases. Subsequently, when

combining these adNNs, difficulties were encountered due to the presence of some oscillations. This led to the use of a single adNN that provides the controller gains for all the loops with similar results.

5. Energy consumption of the integrated Neuro EMC system proves to be slightly lower than the BC. This was due to the pre-training sustained by the reset temperature algorithms. The set points for both the supply air and water temperatures were higher, lowering the demand in the heat pump and transferring it to the pump and the fans. Further improvements are expected under continuous simulation
6. In order to give a wider range of operation to the adNNs in the simulation studies, a simpler algebraic equation for the tube temperature was used. This allowed the systems to be compared on more similar grounds and to provide controller gains without scaling factors.

6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

1. Although the overall energy consumption of the system was not considerably improved, the implementation of non adaptive and adaptive neural networks could still be an excellent way of implementing EMC in real buildings. When comparing the results obtained in this thesis between algorithms that require a thorough knowledge of and measurements in the systems, and those of the NNs, it is clear that the amount of information required to implement the latter is greatly reduced. Moreover, the information required by the NNs is all readily available through the usual monitoring sensors installed in real buildings.
2. The start and stop lead time NNs display good prediction in the times without compromising the comfort of the occupants. The predictions were made using only 10 days of observations in the settling times of the system. Operator knowledge could also play a key component in achieving good selections of the initial training data. The savings obtained are not conspicuous due to the late start of the BC algorithm; nonetheless the over all performance of these NNs presents them as a clear alternative to be considered when integrating this kind of system. It was also noticeable that the performance of the adNNs improved with the number of simulation runs and the amount of data acquired. Therefore the general performance of the system would be expected to improve with time.
3. The reset temperature strategy implemented by means of adNNs gave good energy savings with minimal knowledge of the system. A simple algorithm to

provide an initial training data set is sufficient to produce sensible and reliable results.

4. The Neuro PI gain selector did not produce the best results. The adaptive PI-Neuro controllers developed improved the tracking performance from those of the non adaptive NNs. However the system tended to keep the integral controller gain constant which produced some undesired oscillatory behaviour. This may have been translated into higher energy consumption. Due to difficulties encountered in combining separate NNs, a single NN was used. The difficulties could have been caused by the stiffness of the model equations. However the simplification of the model did not greatly improve the performance of the NNs.
5. Comparing the Neuro PI gain selectors to the fixed controller gains, the results obtained were not improved in tracking performance or in under or over shoots. However when changing seasons, or initial conditions, the performance of the fixed controller gains decays while the NNs does not vary. Still a detailed designing process of the NN is suggested in order to improve these results.

6.2 Future Work

1. It is recommended to implement online optimizers in key variables to improve the ongoing training of the adNNs, producing nearly optimal solutions.
2. It is proposed for further studies to implement the Neuro – EMS developed in this work first with data from the sensors of a real system to test and corroborate the results obtained, and then real time implementation. This would be the software testing stage.

7 REFERENCES

Ahmed, O., Mitchell, J. and Klein, S. Feed Forward Feed Back Controller Using General Regression Neural Network (GRNN) for Laboratory HVAC Systems: Part I- Pressure Control. ASHRAE Transactions. 1998; pp. 613-625.

Ahmed, O., Mitchell, J. and Klein S. Feed Forward Feed Back Controller Using General Regression Neural Network (GRNN) for Laboratory HVAC Systems: Part II- Temperature Control - Cooling. ASHRAE Transactions. 1998; pp. 626-634.

---. Feed Forward Feed Back Controller Using General Regression Neural Network (GRNN) for Laboratory HVAC Systems: Part III- Temperature Control - Heating. ASHRAE Transactions. 1998; pp. 635-643.

Alessandri, A., Bini Verona, F., Parisini, T. and Torrini, A. Neural Approximations for the Optimal Control of Heating Systems. Proceedings on the 3rd IEEE Conference on Control Applications. 1994; Vol. 3.

Anderson, M., Young, P., Hittle, D., Anderson, C., Tu, J. and Hodgson, D. MIMO Robust Control for Heating, Ventilating and Air Conditioning (HVAC) Systems. Proceedings of the 41 st IEEE Conference in Decision and Control. 2002; Vol. 1, pp.167-172.

Angelov, P., Buswell, R., Hanby, V. and Wright, J. A Methodology for Modeling HVAC Components Using Evolving Fuzzy Rules. 26th Annual Conference of the IEEE. 2000; Vol.1, pp. 247-252.

Arguello, B. and Velez, M. Design of a Non Linear HVAC Control System with Thermal Load Estimation. Proceedings of the 4th IEEE Conference in Control Applications. 1995; pp. 33-39.

Arima, M., Hara, E. and Katzberg, J. A Fuzzy Logic and Rough Sets Controller for HVAC Systems. IEEE Proceedings in Communication, Power and Computer. 1995; Vol. 1, pp. 133-138.

ASHRAE, Standard 62. Ventilation for Acceptable Indoor Air Quality. ASHRAE; 1982.

Betta, G. and Pietrosanto, A. Comparison Between IFDI Schemes Based on Expert Systems and Neural Networks. IEEE Instrumentation and Measurement Technology Conference. 1998; Vol. 1.

Cho, S. H. and Zaheeruddin, M. Predictive Control of Intermittently Operated Radiant Floor Heating Systems. Energy Conversion and Management. 2003; Vol. 44.

Chow, M. and Teeter, J. Reduced-Order Functional Link Neural Network for HVAC Thermal System Identification and Modeling. IEEE International Conference in Neural Networks June 1997. 1997; Vol. 1; pp.19-12.

Cichocki, A. and Unbehauen, R. Neural Networks for Optimization and Signal Processing; 1993.

Colmenares, G. and Perez, R. A Reliable Method to Reduce Observations and Variables when Building Neural Network Models. Proceedings of the IV Congreso Interamericano De Computacion Aplicada a La Industria De Proceso. 1999; pp. 1-16.

Cooper, J. and Warwick, K. Output Envelope Adjustment for an Environmental Fuzzy Logic Controller. 3rd IEEE Conference on Control Applications. 1994; Vol. 3, pp. 1625-1627.

Curtiss, P. S., Brandemuehl, M. J. and Kreider, J. F. Energy Management in Central Plants using Neural Networks. ASHRAE Transactions. 1994; Vol. 100, Part 1, pp. 476-493.

Curtiss, P. S., Kreider, J. F. and Brandemuehl, M. J. Local and Global Control of Commercial Building HVAC Systems Using Artificial Neural Networks. Proceedings of the ASME ISEC. 1993; pp. 3029-3044.

Daryanianand, B. and Norford L. Minimum Cost Control of HVAC Systems Under Real Time Prices. Proceedings of the 3rd IEEE Conference on Control Applications. 1994; Vol. 3, pp. 1855-1860.

Demouth, H. and Beale, M. Neural Network Toolbox for use with Matlab [Web Page]. 2003. Available at: www.mathworks.com.

Dexter, A., Geng, G. and Haves, P. The Application of Self Tuning PID Control to HVAC Systems. IEEE Colloquium on Control in Building Energy Management Systems. 1990; Vol. 4, pp.1-3.

Egilegor, B., Uribe, J., Arregi, G., Pradilla ,E. and Susperregi, L. A Fuzzy Control Adapted by a Neural Network to Maintain a Dwelling within Thermal Comfort. Proceedings of the 3rd International IBPSA Conference. 1997.

Erguo, L. and Jinshou, Y. An Input-Training Neural Network Based Non Linear Principal Component Analysis Approach for Fault Diagnosis. Proceedings on the 4th World Congress on Intelligent Control and Automation. 2002; pp. 2755-2759.

Fargus, R. S. and Chapman, C. A Commercial PI-Neural Controller for the Control of Building Services Plant. IEE, UKACC International Conference on Control, Conference Publication. 1998; publication No 455, pp. 1688-1693.

Fausett, L. Fundamentals of Neural Networks, Architectures, Algorithms and Applications; 1994.

Gallant, S. I. Neural Network Learning and Expert Systems; 1994.

Geng, G. and Geary, G. On Performance and Tuning of PID Controllers in HVAC Systems. Second IEEE Conference on Control Applications. 1993; pp. 819-824.

Hailin, P. and Broberg, H. Internet- Based Monitoring and Controls for HVAC Applications. 2002; Vol. 2, Issue 1, pp. 49-57.

Hamdi, M. and Lachivier, G. A Fuzzy Control System based in the Human Sensation of Thermal Comfort. IEEE International Conference on Fuzzy Systems Proceedings. 1998; Vol. 1, pp. 487-492.

Haves, P. and Berkeley, L. Use of Whole Building Simulation in On-Line Performance Assessment: Modeling and Implementation Issues. Seventh International IBPSA Conference. 2001; pp. 335-342.

Hawkings, M., John, R. and Dexter, A. Rule Based Control of Air Flow in Air Conditioning Systems. Control in Building's Energy Management Systems - IEEE Colloquium. 1990; Vol. 5, pp. 1-4.

Haykin, S. Neural Networks, a Comprehensive Foundation. 1999.

He, X. and Asada, H. A New Feedback Linearization Approach to Advanced Control Multi Unit HVAC Systems. Proceedings of the American Control Conference. 2003; Vol. 3, pp. 2311-2316.

Henze, G. and Hindman, R. Control of Air-Cooled Chiller Condenser Fans Using Clustering Neural Networks . ASHRAE Transactions: Research. 2003; Art 4574, pp. 232-244.

Hepworth, S. J. Dexter A. L. (Neural Control of Non-linear HVAC Plant). Proceedings of the 3rd Conference on Control Applications. 1994; Vol. 3.

House, J. and Smith, T. Optimal Control of Building and HVAC Systems . Proceedings of the American Control Conference. 1996; Vol. 6, pp. 4326-4330.

Huaguang, Z. and Cai, L. Decentralized Nonlinear Adaptive Control of an HVAC System . IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews. 2002; Vol. 33, No 4.

Huang, W. Z. Dynamic simulation of Energy Management Control and Functions in VAV-HVAC Systems. 2003.

Incropera, F. Fundamentals of Heat and Mass Transfer; 1996.

Institute for Research in Construction (IRC). National Research Council. Model National Energy Code of Canada for Buildings (MNECCB). 1997.

Jian, W. and Weijian, C. Development of an Adaptive Neuro-Fuzzy Method for Supply Air Pressure Control in HVAC Systems. IEEE International Conference in Systems, Man and Cybernetics. 2000; Vol. 5, pp. 3806-3809.

Kiff, A. and Warwick, K. Distributed Fuzzy Logic for Building Management Systems Using Local Operating Networks. 1996; Vol. 6, Issue 200, pp. 1-4.

Kuntze, H. and Bernard, Th. A New Fuzzy Based Supervisory Control Concept for the Demand Responsive Optimization of HVAC Control Systems. Proceedings of the 37th IEEE Conference on Decision and Control. 1998; Vol. 4, pp. 4258-4263.

Lea, R., Dohmann, E., Preblisky, W. and Jani, Y. An HVAC Fuzzy Logic Zone Control System and Performace Results. Proceedings on the 5th IEEE International Conference. 1996; Vol. 6, pp. 2175-2180.

Li, X., Vaezi-Nejad, H. and Visier, J. A Connectionist Approach to the Development of a Fault Detection and Diagnosis Method for Hydraulic Heating Systems. IEEE International Conference on Systems, Man and Cybernetics. 1996; Vol. 4, pp. 2493-2497.

Liu, Z., Xie, W. and Huang, Y. The energy Control of Air Conditioning Systems in Intelligent Buildings. Proceedings of the 4th World Congress on Intelligent Control and Automation, June 10- 14. 2002.

Mathews, E. and Botha, C. Improved Thermal Building Management with the Aid of Integrated Dynamic HVAC Simulation. Building and Environment. 2003; Vol. 38, Issue 12, pp. 1423-1429.

Mathworks. Simulink User Reference [Web Page]. 2002.

McQuiston, F. C. Heating Ventilating and Air Conditioning, Analysis and Design. 2000.

Mendes, N., Oliveira, G. and Araujo, H. Building Thermal Performance Analysis by Using Matlab/Simulink. Seventh International IBPSA Conference. 2001.

Miller, R. C. and Seem, J. E. Comparison of Artificial Neural Networks with Traditional Methods of Predicting Return Time from Night or Weekend Set Back. ASHRAE Transactions. 1991; Vol. 97, Part 2, pp. 500-508.

Mozer, M., Dodier, R. and Anderson, M. The Neural Network House: An Overview. Proceedings of the American Association for Artificial Intelligence Spring Symposium on Intelligent Environments. 1998; pp. 110-114.

Nagai, T. Optimization Method for Minimizing Annual Energy, Peak Energy Demand,

and Annual Energy Cost Through Use of Building Thermal Storage . ASHRAE Transactions on the 2002 Winter Meeting. 2002; Vol. 108, Part 1, pp. 43-53.

Narendra, K. S. and Parthasarathy, K. Gradient Methods for the optimization of Dynamical Systems Containing Neural Networks. IEEE Transactions on Neural Networks. 1991; Vol. 2, No 2.

Nesler, C. Adaptive Control of Thermal Processes in Buildings. Control System Magazine; 1986; Vol. 6, Issue 4, pp. 9-13.

Panke, R. A. Energy Management Systems and Digital Control.; 2002.

Pape, F., Mitchell, J. and Beckman, W. Optimal Control and Fault Detection in Heating, Ventilating and Air Conditioning Systems. ASHRAE Transactions of the 1991 Winter Meeting. 1991; 97, Part 1729-736.

Pargfrieder, J. and Jorgl, P. An Integrated Control System for Optimizing the Energy Consumption and User Comfort in Buildings. IEEE International Symposium on Computer Aided Control Systems Design Proceedings. 2002; pp. 127-132.

Parlos, A., Menon, S. and Atiya, A. An Adaptive State Filtering Algorithm for Systems with Partially Known Dynamics. ASME Transactions. 2002; Vol. 124, pp. 364-374.

Pasini, A., Ameli, F. and Lore, M. Mixing Height Short Range Forecasting Through Neural Network Modeling Applied to Radon and Meteorological Data. 3rd Conference on Artificial Intelligence Applications to the Environmental Science. 2003.

Pingkang, L., Xiuxia, D. and Xuejun, G. Optimized Neural Net Control using Genetic Algorithm for Intermittent System. Proceedings IEEE TENCON'02. 2002; Vol. 3, pp. 1534-1537.

Rahmati, A., Rashidi, F. and Rashidi, M. A hybrid Fuzzy Logic and PID Controller for Control of Non Linear HVAC Systems. IEEE International Conference on Systems, Man and Cybernetics. 2003; Vol. 3, pp. 2249-2254.

Ribeiro, B. and Cardoso, A. A Model Based Neural Network Controller for Process Trainer Laboratory Equipment . Proceedings of the International Conference on Neural Networks and Genetic Algorithms. 1998; pp. 601-605.

Rich, D. G. The Effect of Fin Spacing on the Heat Transfer and Friction Performance, of Multi-Row, Smooth Plate Fin and Tube Heat Exchangers. ASHRAE Transactions. 1973; Vol. 79, Part II.

Roberts, A. and Oak, M. Non Linear Dynamics and Control for Thermal Room Models.

ASHRAE Transactions: Symposia. 1991; Vol. 97, Part 1, pp. 722-728.

Saboksayr, S. H., Patel, R. V. and Zaheeruddin, M. Energy- Efficient Operation of HVAC Systems using Neural Network Based Decentralized Controllers. Proceedings of the American Control Conference. 1995; pp. 4321-4325.

Salsbury, T. and Diamond, R. Performance Validation and Energy Analysis of HVAC Systems using Simulation . Energy and Buildings. 2000; Vol. 32, pp. 5-17.

Schweppe, F. C., Daryanian, B. and Tabors, R. D. Algorithms for a spot price responding residential load controller . IEEE Transactions on Power Systems. 1989; Vol. 4, No2.

Skapura, D. M. Building Neural Networks. 1996.

Song, Q., Hu, W. J. and Zhao, T. N. Robust Neural Network Controller for Variable Air Flow Volume System . IEE Proceedings Control Theory Applications. 2003; Vol. 150, No 2, pp. 112-118.

Sundarajan, N. and Saratchandran, P. Parallel Architectures for Artificial Neural Networks. 1998.

Taylor, J. G. Neural Network Applications. 1992.

Teeter, J. and Chow, M. Application of Functional Link Neural Network to HVAC Thermal Dynamic System Identification . IEEE Transaction on Industrial Electronics. 1998; Vol. 45, No 1.

Tudoroiu, N. and Zaheeruddin, M. On-Line PI Neuro-controller Strategy for Improving Discharge Air Temperature (DAT) System Performance. Energy Conversion and Management. 2004; Vol. 45, No15-16, pp. 2405-2415.

Virk, G. and Loveday, D. Model Based Control for HVAC Applications. Proceedings on the 3rd IEEE Conference on Control Applications. 1994; Vol. 3, pp.1861-1866.

Virk, G., Loveday, D. and Cheung, J. Model Based Control for BEMS. International Conference on Control. 1994; Vol. 2, pp. 901-905.

Wang, Q., Hang, C., Zhang, Y. and Bi, Q. Multivariable Controller Auto-Tuning with its Applications in HVAC Systems. Proceedings of the American Control Conference. 1999; Vol. 6, pp. 4353-4357.

Wang, Y., Shi, Z. and Cai, W. Total Optimal Operation for HVAC System with Heat Source and Distribution System, PID Auto Tuner and its Application in HVAC Systems . Proceedings of the American Control Conference. 2001; Vol. 3, pp. 2192-2196.

Warwick, K., Irwin, G. D. and Hunt, K. J. Neural Networks for Control and Systems. 1992.

Watton, A., Marcks, R. and Solem, G. Improved Instructional Techniques in HVAC Control Systems. 26th Frontiers in Education Conference. 1995; Vol. 1, pp. 35-38.

Wei, G., Liu, M., Sakurai, Y., Claridge, D. and Turner, W. Practical Optimization of Full Thermal Storage Systems Operation. ASHRAE Transactions. 2002; Vol. 108, Part 2, pp. 360-368.

Wen, J. and Smith, Th. Effect of Thermostat Time Constant Temperature Control and Energy Consumption. IEEE Conference in Sensors for Industry. 2001; pp. 252-257.

Wezenberg, M. and Dewe, B. Adaptive Neural Networks for Tariff Forecasting and Energy Management. IEEE International Conference on Neural Networks. 1995.

Yang, I., Yeo, M. and Kim, K. Application of Artificial Neural Networks to Predict the Optimal Start Time for heating Systems in Buildings. Energy Conversion and Management. 2003; Vol. 44, pp. 2791-2809.

Yazdanpanah, M., Semsar, E. and Lucas, C. Minimization of Actuator Repositioning in Delayed Process Using Flexible Neural Networks. Proceedings of the IEEE Conference on Control Applications. 2003; Vol. 1, pp. 331-335.

Yazdanpanah, M., Semsar, E. and Lucas, C. Nonlinear Control and Disturbance Decoupling of an HVAC System Via Feedback Linearization and Back-Stepping . Proceedings of the 2003 Control Applications Conference. 2003; Vol. 1, pp. 646-650.

Ying-guo, P., Hua-guang, Z. and Bien, Z. A Simple Fuzzy Adaptive Control Method and Application in HVAC. Fuzzy System Proceedings in the IEEE World Congress on Computational Intelligence. 1998; Vol. 1, pp. 528-532.

Yoshida, H., Iguchi, Y. and Wang, F. Total Optimal Operation for HVAC System with Heat Source and Distribution System. Advances in Building Technology 2002, Vol. 2, pp. 1297-1304.

Zaheeruddin, M. and Zheng, G. R. A Dynamic Model of a Multizone VAV System for Control Analysis. ASHRAE Transactions. 1994; Vol. 100, Part II.

---. Multistage Optimal Operating Strategies for HVAC Systems. ASHRAE Transactions. 2001, Vol. 107, Part II.

Zmeureanu, R. Prediction of the COP of Existing Rooftop Units Using ANNs and Minimum Number of Sensors, Energy. 2002; Vol. 27, pp. 889 – 904.

APPENDICES

Appendix 1. Nomenclature and Related Values

Symbol	Variable Represented	Value	Units
A	Face area of the coil	0.107	m^2
A_f	Area of the fins	16.1	m^2
Afo	Ratio of the fin area to the total area	0.942	m^2
Ai	Internal area of the tube	0.906	m^2
Ait	Internal area of the tube per unit of length	1.073	m^2
Ao	Total heat transfer area air side per unit of length	20.24	m
A_t	Area of the tube	1.1816	m^2
Cfin	Specific heat of the fin	896	J/kg-K
COP	Coefficient of performance of the system	-	Dim less
*COP*max	Maximum coefficient of performance	3.5	Dim less
Cpa	Average specific heat constant pressure for the air	1005	J/kg-K
Cpw	Average specific heat for water	4189	J/kg-K
Ct	Specific heat of the tube	385	J/kg-K
Cv	Average specific heat constant volume for the air	717	J/kg-K
d	External diameter of the tube	0.01583	m
Dh	Hydraulic diameter	0.0026	m
din	Internal diameter of the tube	0.0122	m
finthick	Fin thickness	0.0001524	m
hit	Heat transfer coefficient water side	3137.34	W/ m^2-K
ho	Heat transfer coefficient air side	88.07	W/ m^2-K
ht	Heat transfer coefficient coil – air	88.07	W/ m^2-K
j	Design parameter	0.0068	Dim less
JP	Design parameter	0.01754	Dim less
k_a	Thermal conductivity of air	0.0257	W/m-k
k_{fin}	Thermal conductivity of the fin	2707	W/m-k

Symbol	Variable Represented	Value	Units
k_w	Thermal conductivity of water	0.587	W/m-K
Lc	Length of the coil	0.84	m
Ma	Mass flux of air	0.588	kg/s
Ma_1	Mass flux of air for zone 1	0.244	kg/s
Ma_2	Mass flux of air for zone 2	0.344	kg/s
Maw	Ratio of mass flux of air	-	Dim less
mfin	Mass of the fins per unit of length	3.68	Kg/m
mt	Mass per unit of length	38.9	Kg/m
Mw	Mass flux of water	0.468	kg/s
Mwa	Ratio of mass flux of water	-	Dim less
p	Neural Network Input	-	-
Pr_a	Prandlt air	0.704	Dim less
Pr_w	Prandlt water	9.134	Dim less
q_{s1h}	Maximum zone 1 cooling load	1280	W
q_{s1l}	Minimum zone 1 cooling load	750	W
q_{s2h}	Maximum zone 2 cooling load	1800	W
q_{s2l}	Minimum zone 2 cooling load	800	W
qz	Cooling load for the zone	3080	W
qz_1	Cooling load for zone 1	1280	W
qz_2	Cooling load for zone2	1800	W
Re	Reynolds of water	9008	Dim less
Re *din*	Reynolds of the internal diameter	9531	Dim less
St	Stanton number	0.00074	Dim less
t	Actual time	-	h
t	Neural Network Targets	-	-
$T\infty,t$	Temperature of the mechanical room	25	°C
Ta	Mean air temperature	-	°C
Ta,in	Temperature entering the coil	26 (design Value)	°C
Th	High temperature	30	°C
t_h	Maximum temperature occurrence time	15	h
Tl	Low temperature	15	°C
t_l	Minimum temperature occurrence time	6	h
To	Outside temperature	30 (max. value)	°C

Symbol	Variable Represented	Value	Units
Tsa	Temperature of supply air	13(design value)	°C
Tt	Temperature of the tube	-	°C
Tw	Mean water temperature	-	°C
Twr	Temperature of water return	11(design value)	°C
Tws	Temperature of water supply	7 (design value)	°C
Tz	Zone temperature	-	°C
Tz_1	Temperature of the zone 1	24 (set point)	°C
Tz_2	Temperature of the zone 2	22(set point)	°C
Tzfin	Expected final temperature of the zone	*Tz + 2*	°C
$\Delta Tmax$	Delta of temperature maximum for the cooling tower	35	°C
Uhp	Power input for the heat pump	-	Dim less
*Uhp*max	Maximum power input	5000	W
Uo	Overall heat transfer coefficient	51.06	W/ m^2-K
Vface	Velocity of air	4	m/s
Vtank	Volume of the storage tank	0.1735	m^3
V_w	Water velocity	1	m/s
Vz	Total volume of the Zones	55.44	m^3
Vz_1	Volume of zone 1	27.72	m^3
Vz_2	Volume of zone 2	27.72	m^3
x_a	Fin height	0.03175	m
x_b	Fin width	0.0275	m
xv	Percentage of outside air in the building	0.15	Dim less
α	Ratio between the total heat transfer area and the total volume of the coil	800	N/m
γ	Specific heat ratio for air	1.4	Dim less
η_o	Overall efficiency	0.8243	Dim less
η_s	Fin efficiency	0.8135	%
μ_w	Water viscosity	0.00128	N-s/m^2
ρw	Density of water	1000	Kg/ m^3
σ	Ratio of total face area to total area of the coil	0.54	Dim less

Appendix 2. Set Point Selected Via Outside Temperature

These tables present the selected simulated values for determining the interrelated set points. The set point temperature of the zones is the same, and maximum load is assumed.

18	18.65	0.244	18.38	0.344	8	0.4999	12.5	12.5	10.87	11.09	0.3859
19	19	0.2296	19	0.3092	8	0.4811	12.5	12.5	10.99	11.17	0.3529
20	20	0.1964	20	0.2653	8	0.4613	12.5	12.5	11.03	11.13	0.3295
21	21	0.1709	21	0.2318	8	0.4455	12.5	12.5	11.03	11.06	0.3142
18	18.65	0.244	18.38	0.344	7.5	0.5139	12.5	12.5	11.14	11.47	0.3042
19	19	0.2296	19	0.3092	7.5	0.4948	12.5	12.5	11.27	11.58	0.2795
18	18.65	0.244	18.38	0.344	7	0.5285	12.5	12.5	11.31	11.77	0.2571
20	20	0.1964	20	0.2653	7.5	0.4746	12.5	12.5	11.38	11.58	0.2566
18	18.65	0.244	18.38	0.344	6.5	0.5438	12.5	12.5	11.41	12.01	0.2256
19	19	0.2296	19	0.3092	7	0.509	12.5	12.5	11.45	11.86	0.2369
18	18.65	0.244	18.38	0.344	6	0.5598	12.5	12.5	11.47	12.21	0.2025
20	20	0.2104	20	0.2843	6.5	0.5072	13	13	12.3	12.91	0.1738
21	21	0.1709	21	0.2318	7.5	0.4584	12.5	12.5	11.47	11.58	0.2394
20	20	0.2104	20	0.2843	7	0.4927	13	13	12.22	12.7	0.1931
20	20	0.2266	20	0.3062	7	0.4977	13.5	13.5	12.74	13.41	0.1782
18	18.65	0.244	18.38	0.344	5.5	0.5766	12.5	12.5	11.5	12.38	0.1847

20	20	0.2194	20	0.2843	7.5	0.4788	13	13	12.1	12.45	0.2194
20	20	0.2266	20	0.3062	7.5	0.4837	13.5	13.5	12.68	13.21	0.1976
22	22	0.1508	22	0.2053	7.5	0.4451	12.5	12.5	11.54	11.58	0.2262
20	20	0.2104	20	0.2843	8	0.4655	13	13	11.9	12.14	0.2583
20	20	0.2266	20	0.3062	8	0.4703	13.5	13.5	12.57	12.98	0.2236
20	20	0.2104	20	0.2843	8.5	0.4527	13	13	11.6	11.74	0.3254
20	20	0.2266	20	0.3062	8.5	0.4574	13.5	13.5	12.41	12.69	0.2614
20	20	0.2116	20	0.2859	9	0.4408	13	13.04	11.16	11.23	0.2393
20	20	0.2266	20	0.3062	9	0.445	13.5	13.5	12.15	12.33	0.2042
19	19	0.2296	19	0.3092	6.5	0.5239	12.5	12.5	11.56	12.11	0.2083
20	20	0.1964	20	0.2653	7	0.4883	12.5	12.5	11.61	11.91	0.2162
19	19	0.2296	19	0.3092	6	0.5394	12.5	12.5	11.63	12.31	0.1872
21	21	0.1816	21	0.2463	7	0.4748	13	13	12.42	12.77	0.1771
21	21	0.1709	21	0.2318	7	0.4719	12.5	12.5	11.75	11.95	0.2001
20	20	0.1964	20	0.2653	6.5	0.5028	12.5	12.5	11.76	12.19	0.1894
21	21	0.1816	21	0.2463	8	0.4484	13	13	12.01	12.15	0.2393
21	21	0.1937	21	0.2627	8	0.4516	13.5	13.5	12.75	13.03	0.2042
21	21	0.2075	21	0.2814	8	0.4553	14	14	13.34	13.8	0.1834
21	21	0.1816	21	0.2463	8.5	0.436	13	13	11.63	11.69	0.3066
21	21	0.1937	21	0.2627	8.5	0.4391	13.5	13.5	12.54	12.72	0.2401

21	21	0.2075	21	0.2814	8.5	0.4427	14	14	13.23	13.56	0.2073
21	21	0.2235	21	0.3031	8.5	0.4469	14.5	14.5	13.79	14.31	0.1876
21	21	0.2421	21	0.3284	8.5	0.4518	15	15	14.26	14.98	0.1747
21	21	0.1937	21	0.2627	9	0.4271	13.5	13.5	12.21	12.3	0.3019
21	21	0.2075	21	0.2814	9	0.4307	14	14	13.05	13.27	0.242
21	21	0.2235	21	0.3031	9	0.4348	14.5	14.5	13.69	14.08	0.2113
21	21	0.2421	21	0.3284	9	0.4395	15	15	14.22	14.8	0.1926
20	20	0.1964	20	0.2653	6	0.5178	12.5	12.5	11.86	12.42	0.1701
22	22	0.1508	22	0.2053	7	0.4582	12.5	12.5	11.88	11.98	0.1873
22	22	0.1592	22	0.2167	7	0.4603	13	13	12.6	12.84	0.1643
21	21	0.1709	21	0.2318	6.5	0.4859	12.5	12.5	11.95	12.25	0.1715
22	22	0.1592	22	0.2167	7.5	0.4471	13	13	12.39	12.53	0.188
22	22	0.1686	22	0.2294	7.5	0.4494	13.5	13.5	13.08	13.36	0.1661
21	21	0.244	21	0.2318	6	0.5006	12.5	12.5	12.08	12.5	0.1566
22	22	0.1592	22	0.2167	8	0.4344	13	13	12.1	12.14	0.2245
22	22	0.1686	22	0.2294	8	0.4967	13.5	13.5	12.9	13.07	0.1891
22	22	0.1791	22	0.2438	8	0.4392	14	14	13.54	13.88	0.1684
22	22	0.1592	22	0.2167	8.5	0.4223	13	13	11.63	11.61	0.2942
22	22	0.1686	22	0.2294	8.5	0.4245	13.5	13.5	12.64	12.72	0.2239
22	22	0.1791	22	0.2438	8.5	0.4269	14	14	13.39	13.61	0.1909
22	22	0.191	22	0.26	8.5	0.4297	14.5	14.5	14	14.39	0.1711
22	22	0.1686	22	0.2294	9	0.4128	13.5	13.5	12.23	12.24	0.2865
22	22	0.1791	22	0.2438	9	0.4152	14	14	13.17	13.29	0.2241
22	22	0.191	22	0.26	9	0.4179	14.5	14.5	13.87	14.14	0.1932
22	22	0.2047	22	0.2788	9	0.421	15	15	14.45	14.9	0.1744
22	22	0.1508	22	0.2053	6.5	0.472	12.5	12.5	12.11	12.31	0.1629
23	23	0.1413	23	0.193	7	0.4481	13	13	12.76	12.89	0.1538

23	23	0.1413	23	0.193	7.5	0.4351	13	13	12.52	12.55	0.1767
23	23	0.1487	23	0.2032	7.5	0.4367	13.5	13.5	13.25	13.42	0.1549
23	23	0.1346	23	0.1838	6.5	0.4601	12.5	12.5	12.27	12.35	0.153
23	23	0.157	23	0.2145	8	0.426	14	14	13.73	13.94	0.1464
23	23	0.157	23	0.2145	8.5	0.414	14	14	13.54	13.65	0.1778
23	23	0.1662	23	0.2271	8.5	0.4159	14.5	14.5	14.2	14.46	0.1581
23	23	0.157	23	0.2145	9	0.4026	14	14	13.26	13.29	0.2102
23	23	0.1662	23	0.2271	9	0.4044	14.5	14.5	14.03	14.19	0.179
23	23	0.1766	23	0.2413	9	0.4065	15	15	14.66	14.98	0.1602
24	24	0.1266	24	0.1737	7	0.4375	13	13	12.92	12.93	0.1449
24	24	0.1327	24	0.1819	7.5	0.4259	13.5	13.5	13.41	13.47	0.1455
24	24	0.1393	24	0.191	8	0.4149	14	14	13.9	14	0.1464
24	24	0.1393	24	0.191	8.5	0.4032	14	14	13.67	13.68	0.1671
24	24	0.1466	24	0.2011	9	0.3932	14.5	14.5	14.18	14.23	0.1675
25	25	0.1144	25	0.1575	6.5	0.4413	13	13	13.3	13.3	0.1223

From this data two algorithms were developed. The first one was the outdoor air temperature based set point selector, showed in chapter 3, and the second one was a zone temperature based set point selector, displayed in equations (23a) and (23b).

$$Tsaset = 0.214 \cdot Tzset + 8.7746 \quad (23a)$$

$$Twsset = 0.1598 \cdot Tzset + 4.3687 \quad (23b)$$

Tzone set	*Tsa set*	*Tws set*
18	12.63	7.25
19	12.84	7.41
20	13.06	7.57
21	13.27	7.73
22	13.48	7.88
23	13.70	8.04
24	13.91	8.20
25	14.13	8.36

Table A-1. Estimation of the set points as function of the Zone set point

For the energy comparison between the three algorithms: Outdoor air temperature based (*To* based), zone temperature based (*Tz* based) and cooling or heating load based (*qs* based), two cases were considered: Random conditions and a normal 24h simulation run. The first four figures compare the performance under random set points and cooling loads. The following four figures show the comparison of the three set point selection algorithms under normal 24h simulation run. The performance of the algorithms diverges from condition to condition, and case to case, however the *To* based algorithm, consistently produce lower energy consumption than the *Tz* based algorithm, reason why this later was not chosen for the base case in this project. Between the *To* based and the *qs* based, energy consumption was not the main reason for the selection. Since one of the objectives of the BC in this project is to generate a suitable training data set for the NN, the *qs* based was bound to be eliminated. To avoid generating a data set that is dependent on the simulated outdoor air temperature profiles, the random case was often used in this project for the NNs training data sets, rendering a problem in implementing the *qs* based algorithm. Furthermore, the difference in energy consumption between the *To* based and the *qs* based were small enough to be omitted. The energy difference for both evaluated cases is

displayed in *Figure A.9*. If this system is ever implemented in a real building, the operator will play an important role in data acquisition for fitting the polynomial.

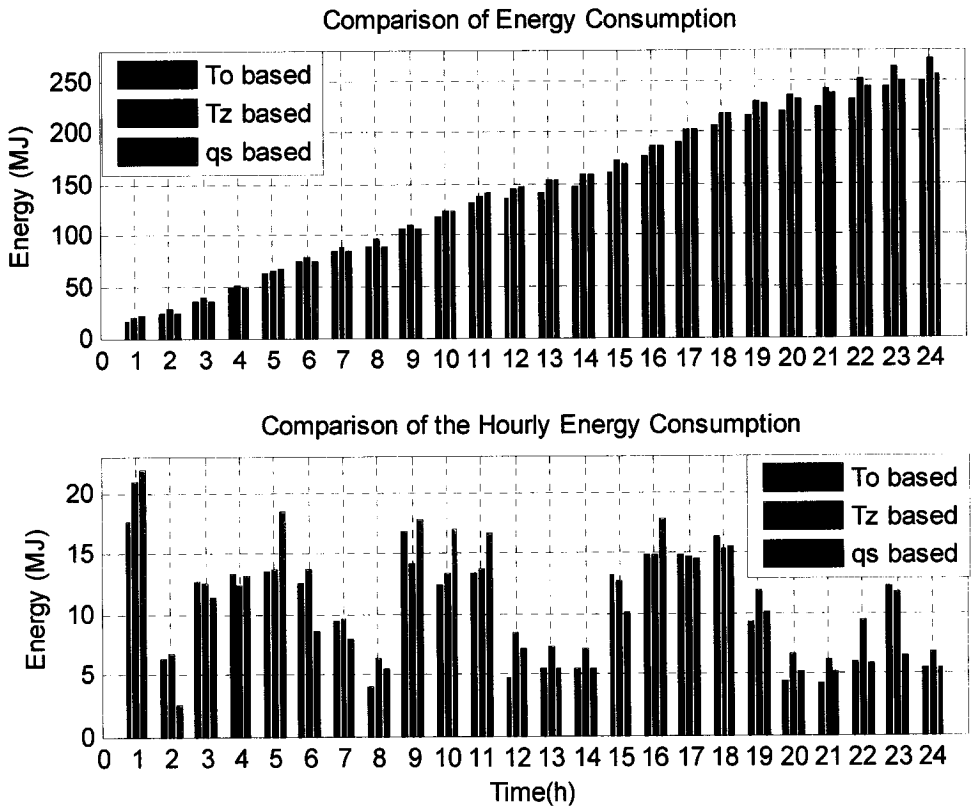


Figure A.1. Set Point Selection Energy Comparison, Random Data .

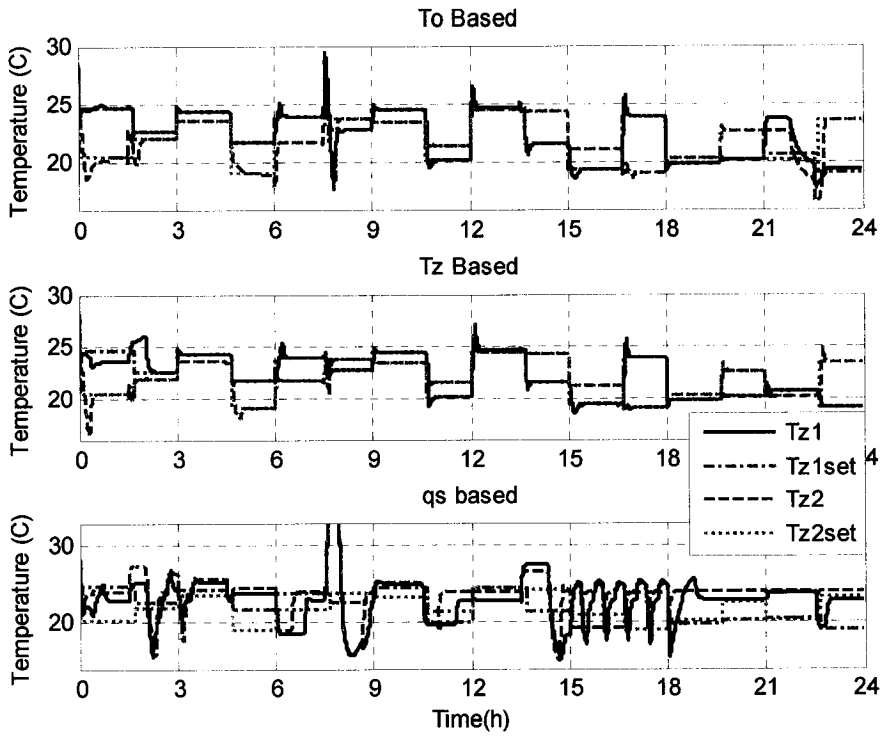


Figure A.2. Comparison of Set Point Selection, Random Winter Conditions.

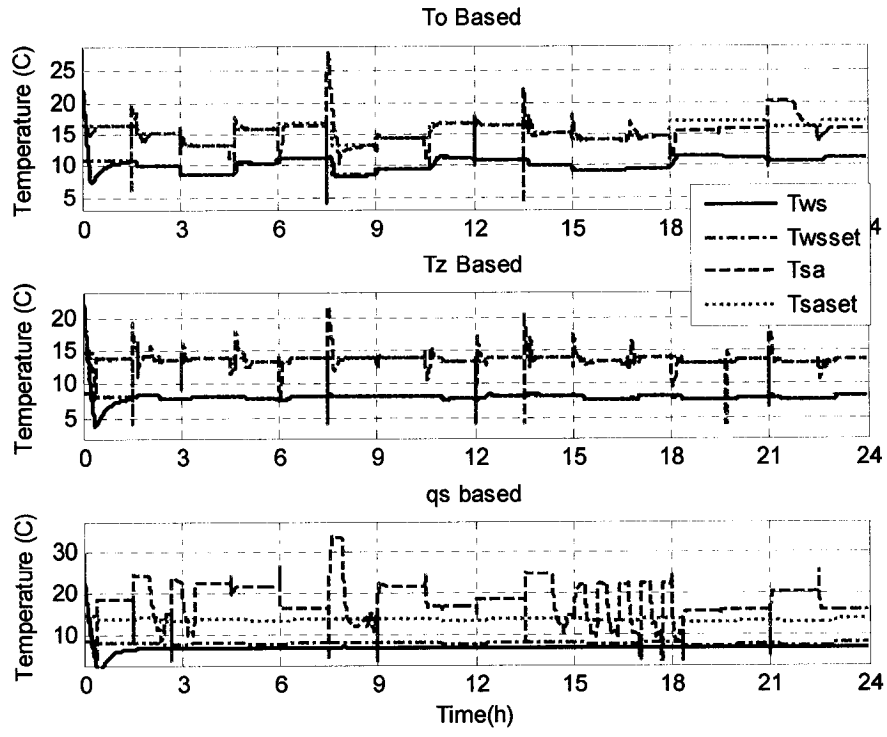


Figure A.3 Comparison of Set Point Selection, Random Summer Conditions.

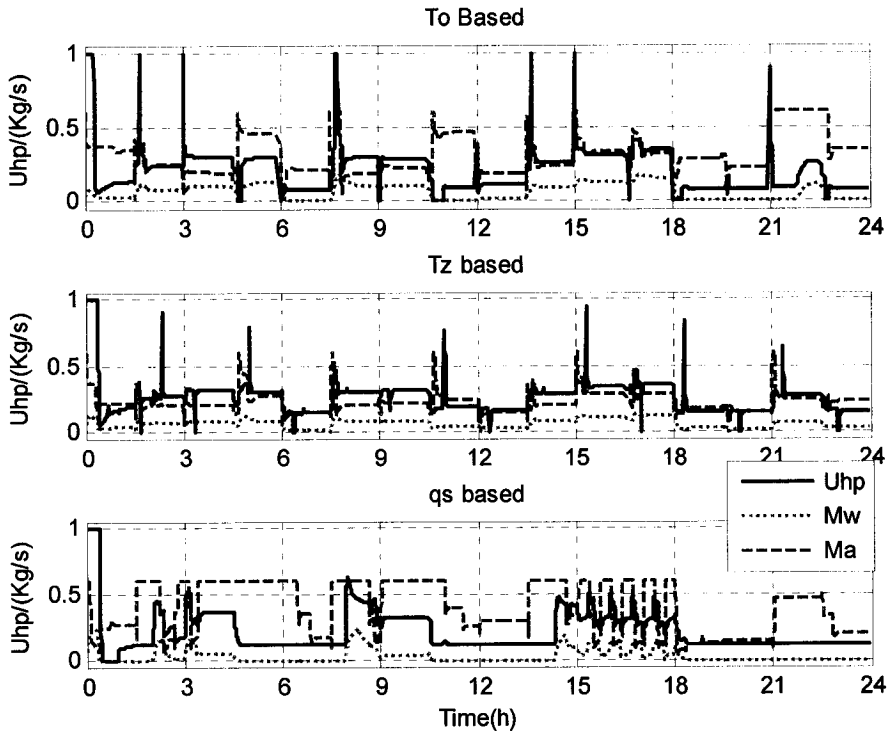


Figure A.4. Mass Flow Rates of Air and Water, Power Input.

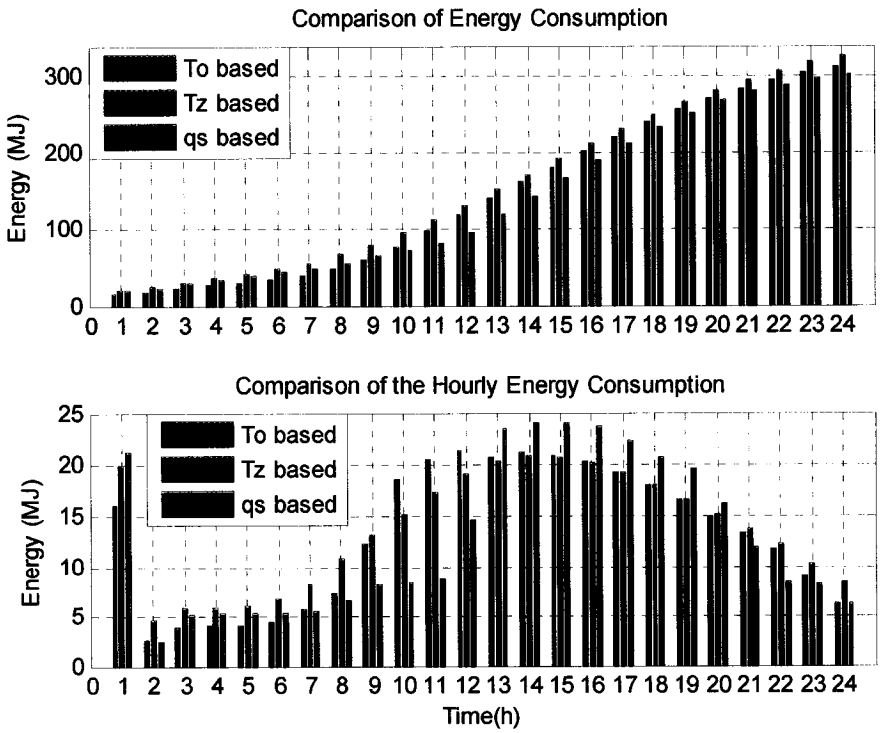


Figure A.5. Energy Comparison Normal 24h Simulation Run .

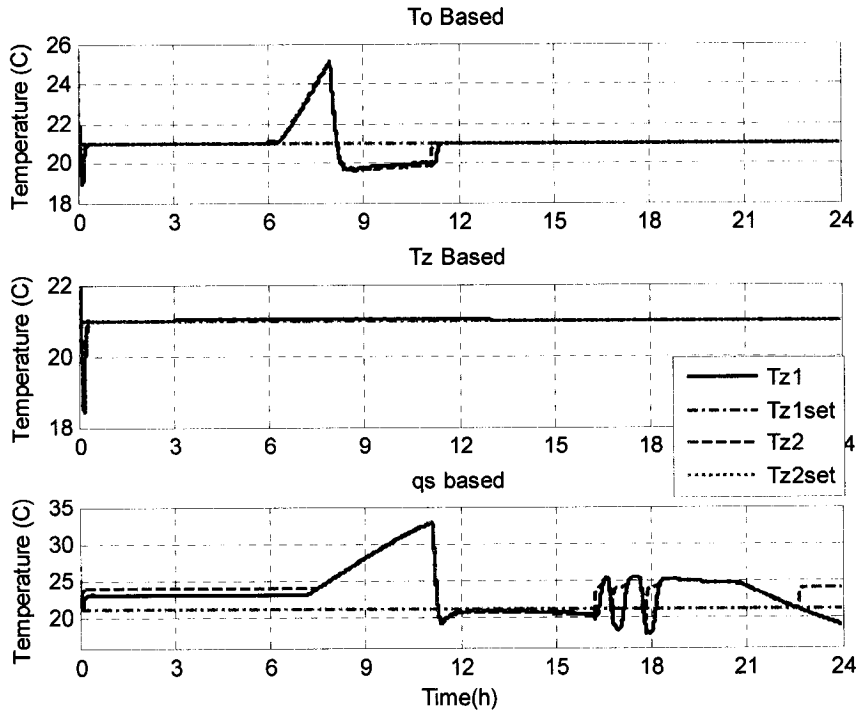


Figure A.6. 24h Simulation Run, Winter Conditions.

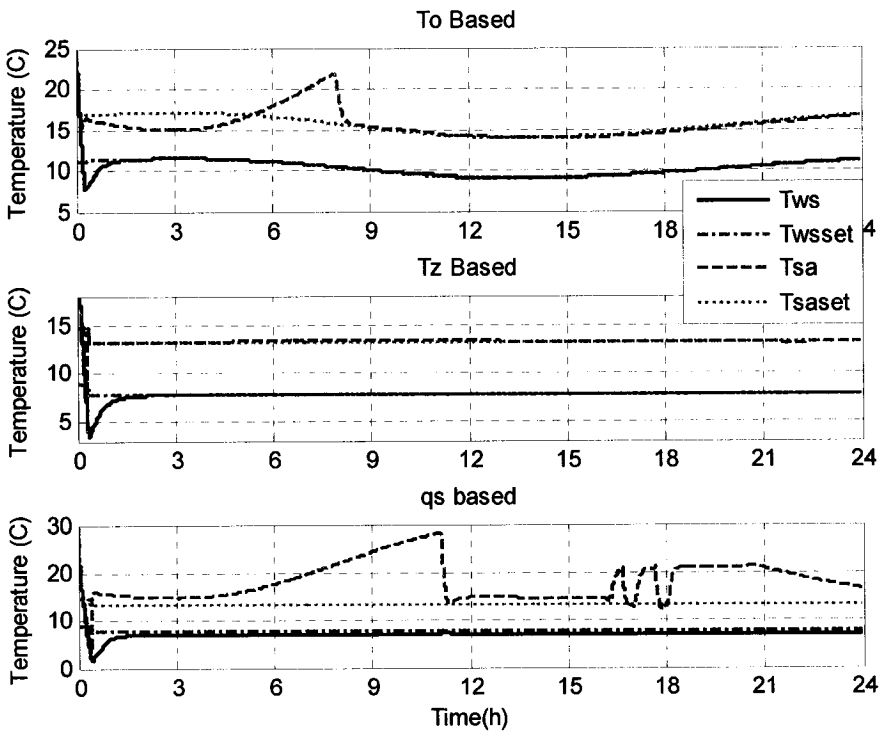


Figure A.7. 24h Simulation Run, Summer Conditions.

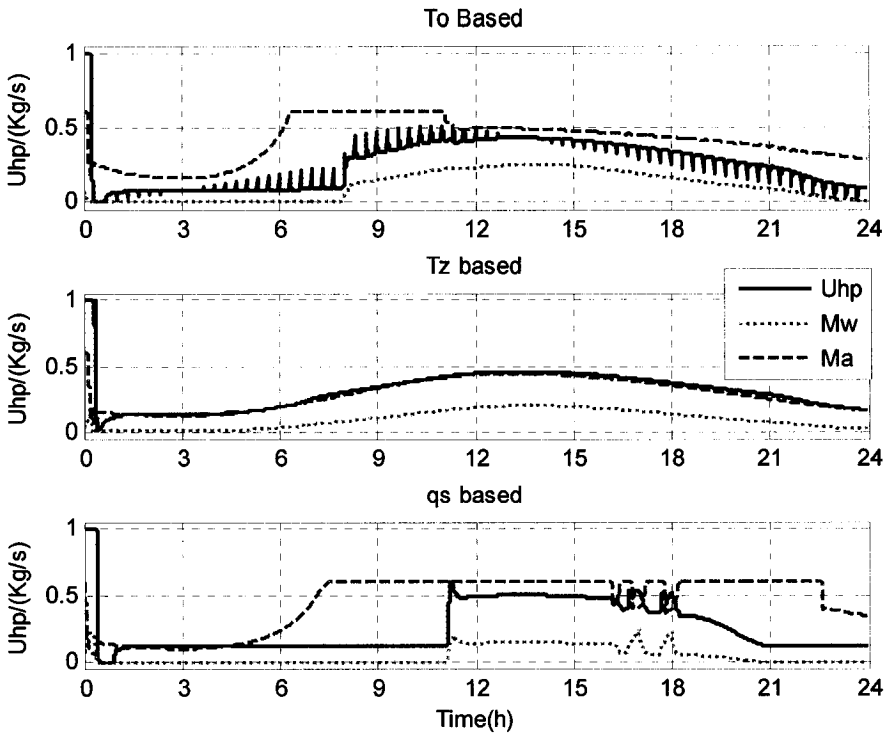


Figure A.8. Mass Flow Rate of Air and Water and Power Input.

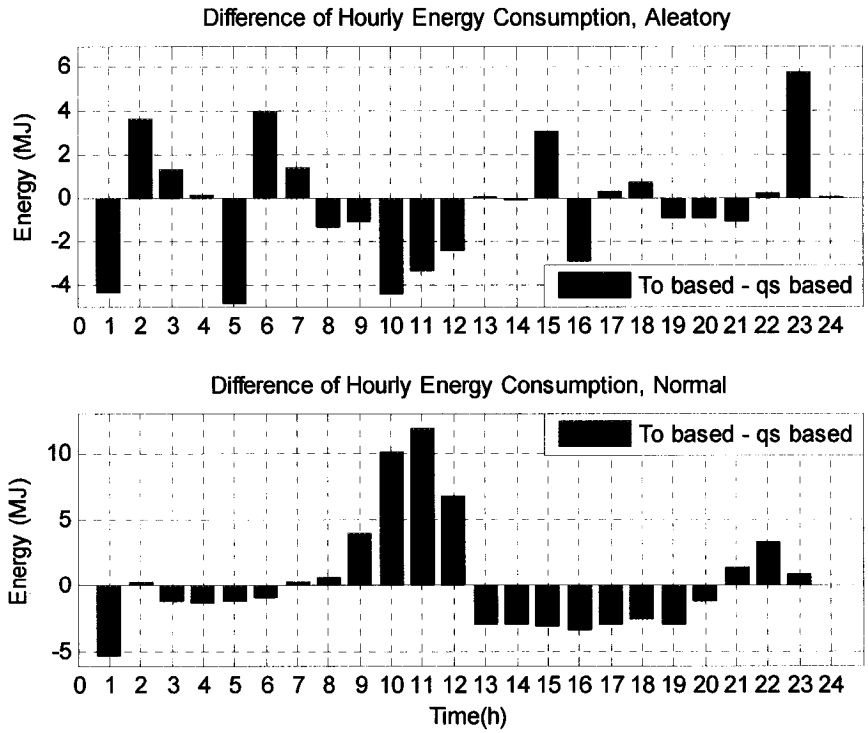


Figure A.9. Energy Difference Between the To based and qs based .