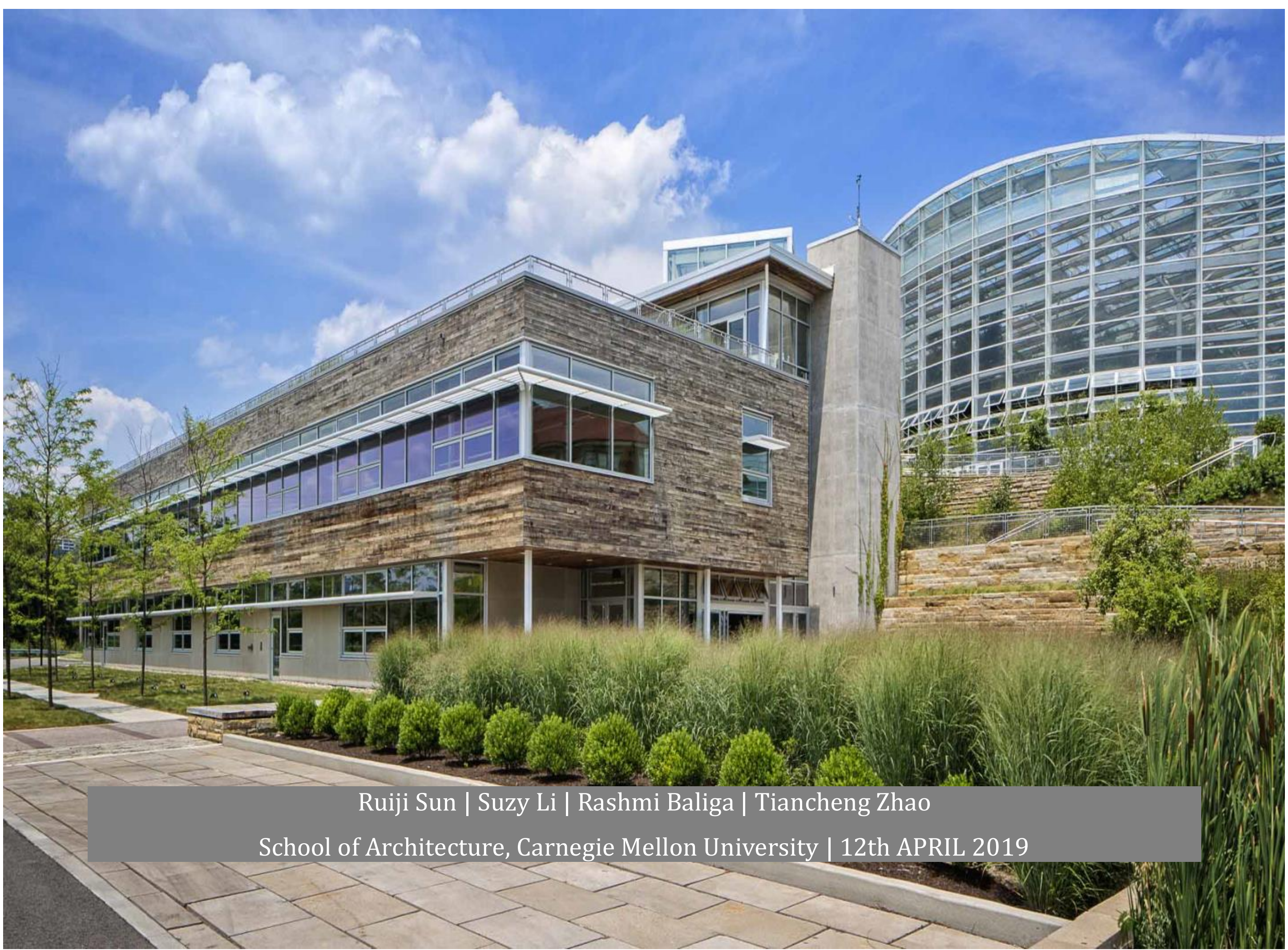
Control System Design and Simulation for Building Systems - Part 1: Short Cycle Issue

Centre for Sustainable Landscapes Building

Part2: Participatory Voting Control for Thermal Comfort (Page 61-Page150)



Ruiji Sun | Suzy Li | Rashmi Baliga | Tiancheng Zhao

School of Architecture, Carnegie Mellon University | 12th APRIL 2019

Executive Summary

The aim of this study was to conduct a detailed analysis of the provided HVAC control scenario, and to derive and achieve a control objective on the basis of that analysis. The analysis was conducted using a simulation-based building control test bed. The specific scenario provided for the study was “Short-cycling” for the heat pump at the Centre for Sustainable Landscapes building, which had simple on and off control. Given the short cycling limitation of 0.5 hours between consecutive on or off actions, a method to improve the indoor thermal comfort of occupants was the control objective.

Using the existing problem description and baseline control strategy, the state variables, control variables and control objectives were defined. The state variables, used to assess the indoor thermal conditions, were the Indoor Air Temperature (IAT) and the Predicted Mean Vote (PMV). The control variable used to implement a control action on the basis of the state variables was the on/off action of the heat pump. During this study, two control objectives were formulated – The reduction of energy consumption, and an increase in the indoor thermal comfort for occupants.

The baseline control logic was tested using EnergyPlus simulation, and the results from the simulation were post processed using R Studio, where the result data was divided into subsets for each zone. The subsets included data for outdoor Dry Bulb Temperatures (DBT), IAT, setpoint temperature, HVAC electric demand and heating coil power. A new column was added as an indicator of Comfort, based on the ASHRAE 55 Standard, 2010. Different scenarios involving the state variables were coded on python, and the simulation results were post processed in the same way.

Using the results of this analysis, the final control logic arrived at was based on implementing a scheduled night time setback temperature of 12 °C, and changing the PMV constraint to take into account the mean PMV, as opposed to the minimum PMV. This strategy allowed us to achieve our control objective. The results of the new control logic implementation as well as the process leading up to it are discussed in detail in this report.

Table of Contents

1. Introduction

1.1. Objectives

This part I of the study is aiming at designing, implementing and analyzing an HVAC control strategy based on a simulation-based building control test-bed. The specific scenario of the HVAC control is given, a new control strategy will be provided to improve the energy and thermal comfort performance of the HVAC system.

1.2. Scenario

The given scenario is a "Short-cycling" heat pump control in the Center for sustainable landscapes building. The heat pump in the CSL building is a simple on/off heat pump, functioning as the single heating source for all zones. The heat pump has a limitation of short-cycling in 0.5 hour. Once the heat pump is turned off, it can only be turned back on after 0.5 hour. This may have negative impact on indoor thermal comfort.

Control Output

The output is a designed control logic for the on/off short-cycle heat pump.

Control Input

The following observations are provided as needed for control logic input:

- Index 0: hour of the day (0-23)
- Index 1: minute of the hour (0-59)
- Index 2: day of the week (0-6, 0 is Monday)
- Index 3: current outdoor air temperature ($_o$C)
- Index 4: current outdoor air RH (%)
- Index 5: current diffuse solar radiation (W/m2)
- Index 6: current direct solar radiation (W/m2)
- Index 7-16: current IAT of the 10 zones ($_o$C)
- Index 17-26: current IAT heating setpoint of the 10 zones ($_o$C)
- Index 27-36: current PMV of the 10 zones (if the zone is not occupied, then return -999)
- Index 37: DX coil electric demand (W)
- Index 38: HVAC total electric demand (W)

Control timestep: 10 minutes

Control simulation period: Jan 1$_{st}$ – Jan 31$_{st}$

2. Analysis and Approach

2.1. Problem Identification – Baseline Analysis

Baseline Control Strategy

In the baseline, the control variable is the heat pump, and the state variables are the minimum zone indoor air temperature and the minimum PMV. PMV, predicted mean vote, is a seven-point thermal sensation scale (-3 to +3), which predicts the average vote of a large group of people on their thermal sensation from cold to hot. The heat pump is controlled by checking the two state variables. If the minimum PMV of all zones is lower than -0.5 or if the minimum indoor air temperature of all zones is lower than 18 $_o$C, the heat pump will be turned on. However, the control is limited by the short-cycling scenario, where the heat pump must stay off for 30 minutes before it can be turned on again. Thus, sometime even when the temperature is lower than 18 ₀C or the PMV is lower than -0.5, the heat pump still cannot be turned on due to this limitation, until the state lasted for 30 minutes. This is a major constrain for the thermal comfort.

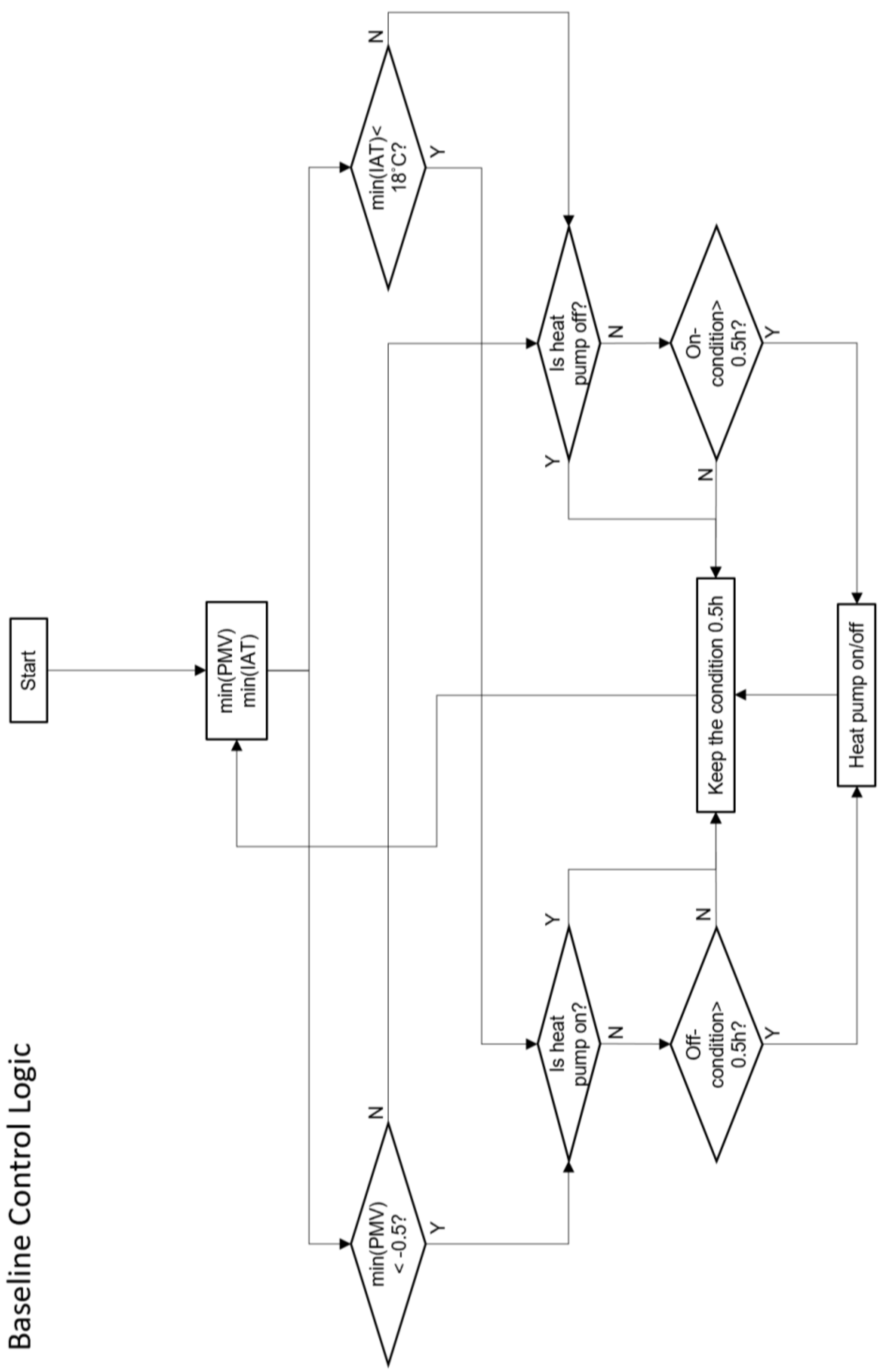


Figure 2.1.1 Baseline Control Logic Flow Chart

Baseline Results Analysis

The EnergyPlus simulation results are analyzed in Rstudio and visualized in DesignBuilder Result Viewer.

Outdoor Climate Condition

The outdoor dry-bulb temperature during the control simulation period is ranging from -16.7 °C to 19.4 °C, average at -0.5 °C. Relative humidity is ranging from 25% to 100%, with the mean of 73.86%, as shown in Table 2.1.1.

Table 2.1.1 Site Outdoor Condition

	out_DBT (oC)	out_RH (%)
Min	-16.7000	**25.00**
1st Q	-5.6000	**64.00**
Median	-2.0583	**77.50**
Mean	-0.5074	**73.86**
3rd Q	2.9500	**86.50**
Max	**19.4000**	**100.00**

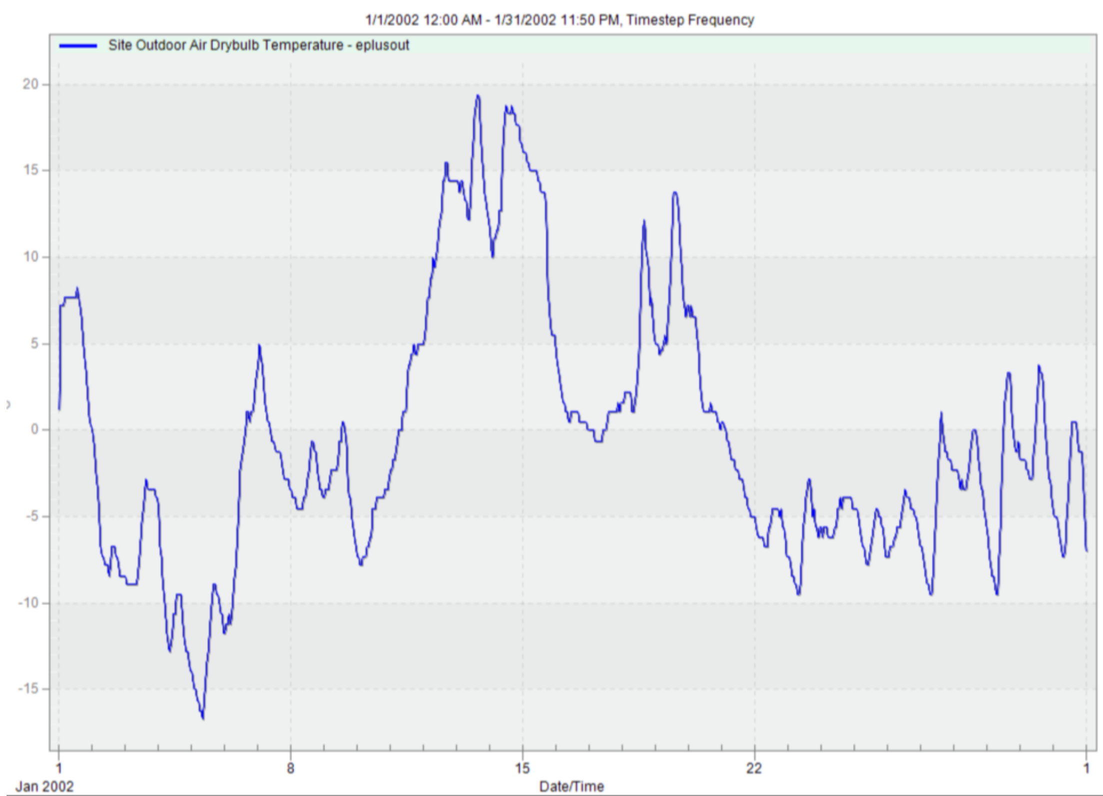


Figure 2.1.2 Site Outdoor Dry-bulb Temperature

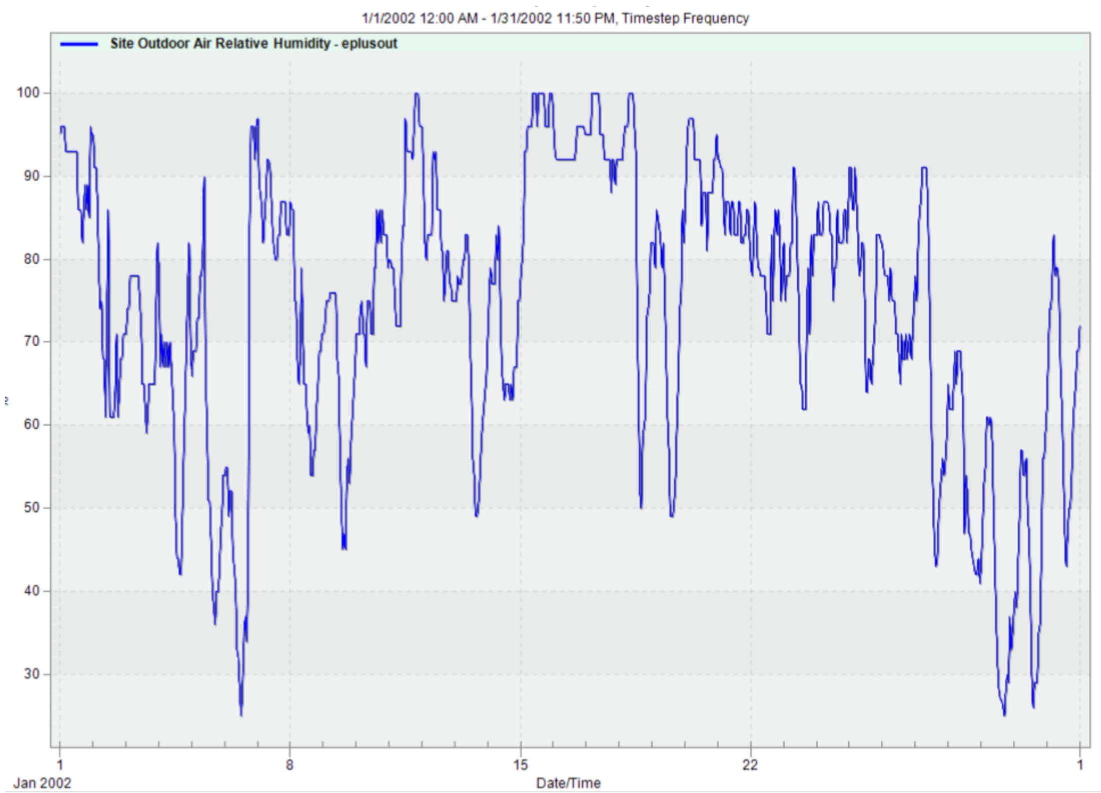


Figure 2.1.3 Site Outdoor Relative Humidity

Zone Air Temperature and Setpoint

The setpoint temperature is 22 °C during 7:00 am to 8:10 pm during weekdays, and the setback temperature for the rest of the time is 18 °C, as shown in Figure 2.1.4. However, the control logic for the baseline is not corresponding with the setpoint in the system. In the control logic, the heat pump is turned on when the lowest zone air temperature is lower than 18 °C for all zones or the lowest PMV is lower than -0.5.

Table 2.1.2 Zone Air Temperature Results and Setpoint

IAT	1_conf	1_class	1_NM	1_SW	1_SE	2_conf	2_SMW	2_SE	2_NMW	2_NW	setpoint
Min	14.18	13.53	15.18	12.32	13.09	13.47	12.30	12.17	14.73	11.90	**18.00**
1st Q	20.23	20.05	20.95	19.56	19.93	19.79	19.61	19.54	20.85	19.36	**18.00**
Median	22.29	22.12	23.51	21.73	22.15	21.86	21.94	21.87	23.41	21.59	**18.00**
Mean	22.01	21.85	23.00	21.44	21.80	21.55	21.58	21.51	22.88	21.31	**19.54**
3rd Q	23.73	23.66	24.96	23.38	23.75	23.29	23.58	23.51	24.90	23.28	**22.00**
Max	**27.63**	**26.93**	**28.01**	**27.53**	**27.70**	**27.15**	**27.63**	**27.62**	**27.90**	**27.26**	**22.00**

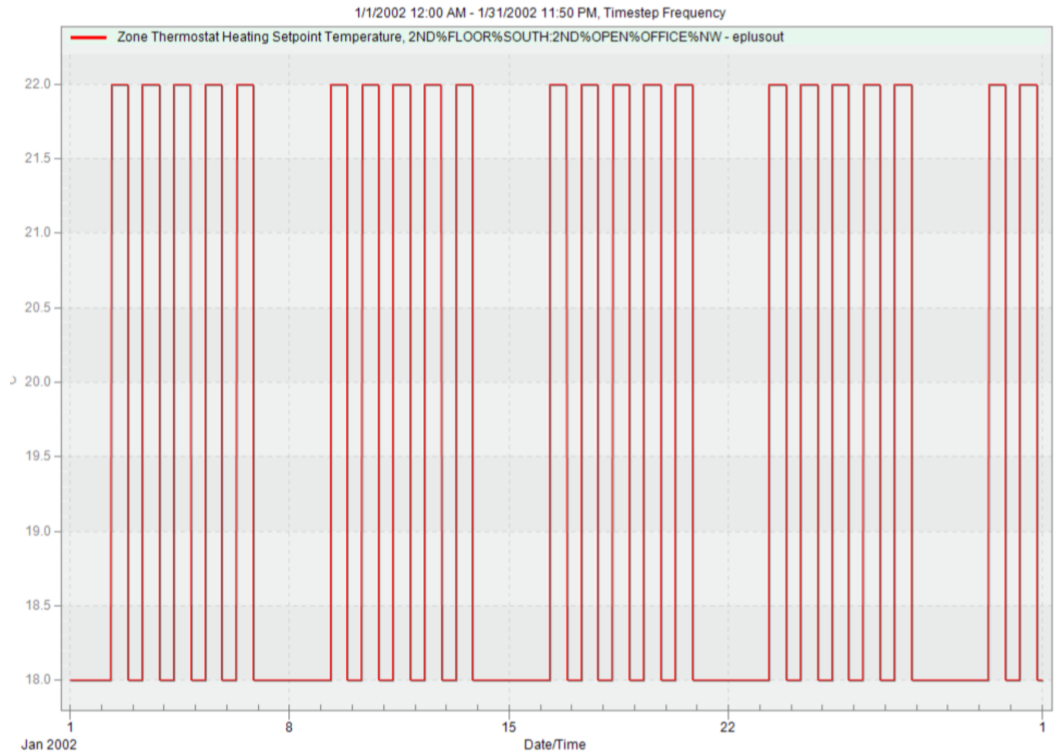


Figure 2.1.4 Setpoint temperature for all zones

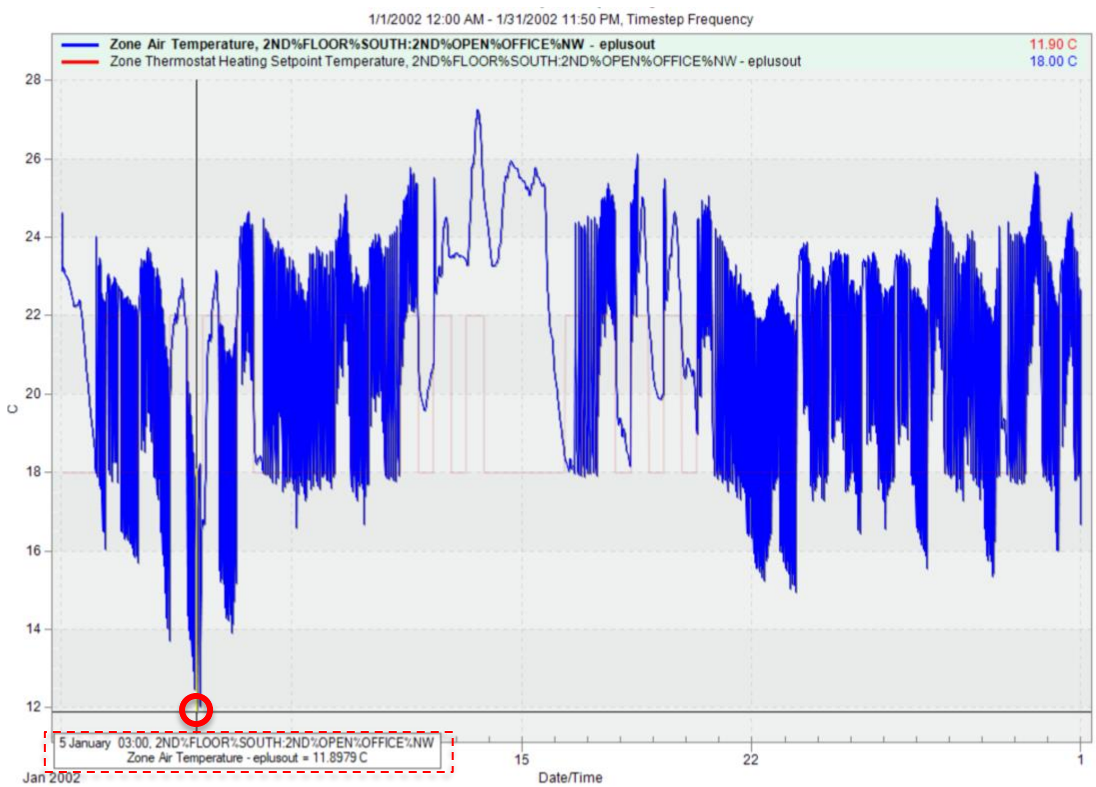


Figure 2.1.5 Minimum Zone Air Temperature

The lowest zone air temperature happened in the open office zone on the Northwest of the second floor, the lowest indoor air temperature is 11.9 °C, happened on Jan 5^{th} at 3am, during unoccupied hours, as indicated in Figure 2.1.5. It corresponds with the lowest outdoor temperature, as shown in Figure 2.1.2.

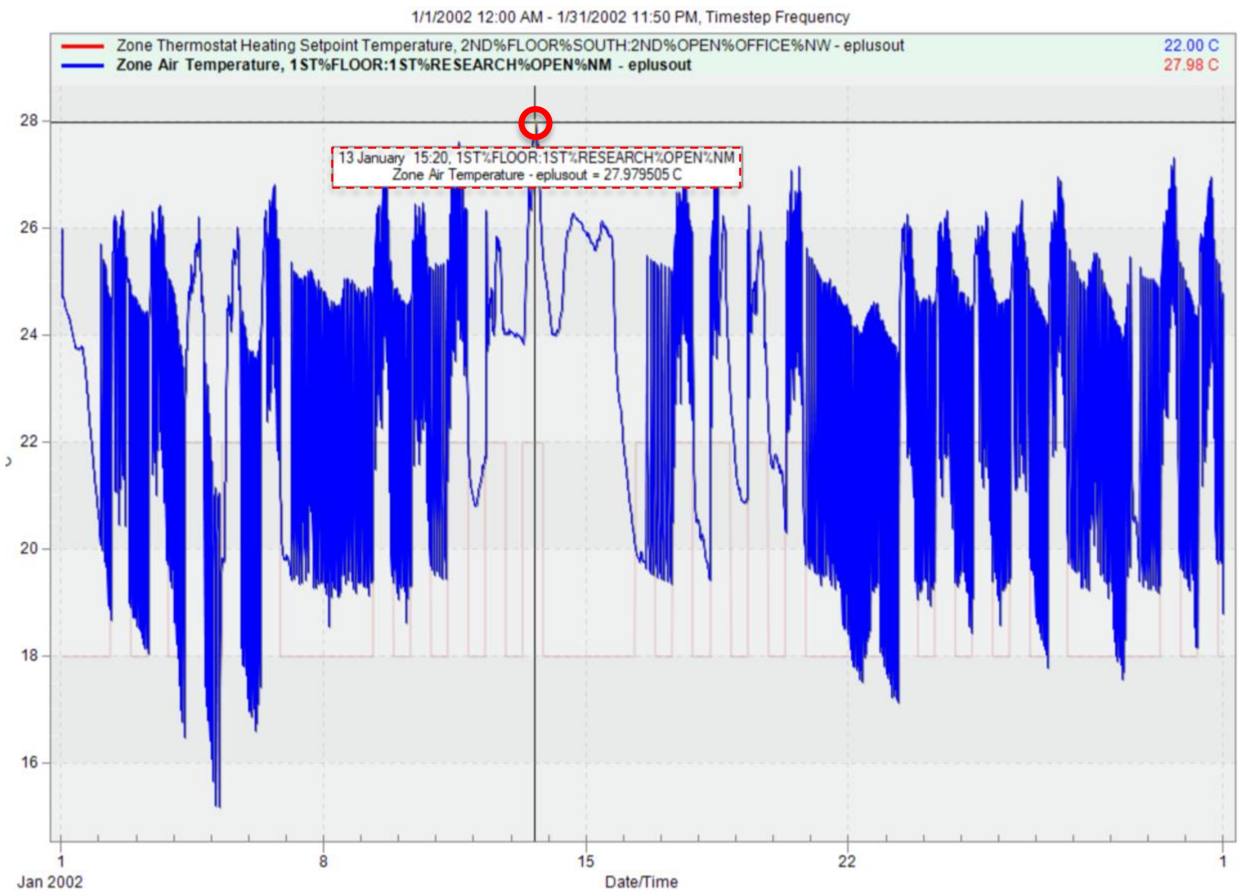


Figure 2.1.6 Maximum Zone Air Temperature

The highest zone air temperature appeared in the same zone as the lowest zone air temperature. The highest indoor air temperature is 28 °C, on Jan 13^{th} at 3:20 pm, during normal office hours, as indicated in Figure 2.1.6. This also correlates to the outdoor temperature condition as the highest point, as shown in Figure 2.1.2. This will have negative impact on the indoor thermal comfort for occupants in this zone. However, in this given scenario, there is no cooling mechanism designed for the space, so this puts a limitation on the overall comfort percentage of the space since the outdoor temperature is above 15 °C during Jan 13^{th} and Jan 15^{th}. The minimum PMV for these two days are above 0.5, as indicated in Figure 2.1.11.

Predicted Mean Vote (PMV) – Thermal Comfort Performance

PMV, as the thermal sensation indicator, is influenced by six parameters, air temperature, mean radiant temperature, relative humidity, air speed, metabolic rate and clothing insulation (Fanger, 1970). The PMV value is set to NA when the space is not occupied. With the number of the NA's in the dataset, it indicates that open office, classroom, and conference room are in different schedules, as shown in Table 2.1.2. The PMV values are ranging from -1.96 to +0.8 in all zones.

Table 2.1.3 PMV results for 10 zones

PMV	1_conf	1_class	1_NM	1_SW	1_SE	2_conf	2_SMW	2_SE	2_NMW	2_NW
Min	-0.967	-1.089	-1.4099	-1.7716	-1.7012	-1.176	-1.9408	-1.9546	-1.4432	-1.9173
1st Q	-0.421	-0.567	-0.2844	-0.5307	-0.4644	-0.626	-0.6577	-0.6707	-0.3631	-0.7866
Median	-0.202	-0.118	-0.0190	-0.2968	-0.2211	-0.425	-0.4225	-0.4336	-0.0763	-0.5205
Mean	-0.229	-0.077	-0.0761	-0.3063	-0.2374	-0.444	-0.4280	-0.4385	-0.1368	-0.5422
3rd Q	-0.063	0.518	0.1251	-0.0760	-0.0062	-0.278	-0.1945	-0.2025	0.0685	-0.3153
Max	0.741	0.654	0.7928	0.7742	0.7949	0.564	0.7209	0.7207	0.7357	0.5541
#of NA	**3887**	**4364**	**3023**	**3023**	**3023**	**3887**	**3023**	**3023**	**3023**	**3023**

Figure 2.1.7 PMV for second floor open office on Southeast

The lowest PMV happened on the 23 Jan, at 8:00 am at the open office located at Southeast on the second floor, as low as -1.96, as indicated in Figure 2.1.8. It's the first hour after scheduled occupant's arrival time. However, it's during the time when the heat pump is under off condition within the 30 minutes short cycle limitation. The zone air temperature is 15 °C at the moment, even though both minimum PMV and minimum indoor air temperature have met the condition to turn on the heat pump, it won't be able to be turned on due to the limitation, which lead to the uncomfortable condition.

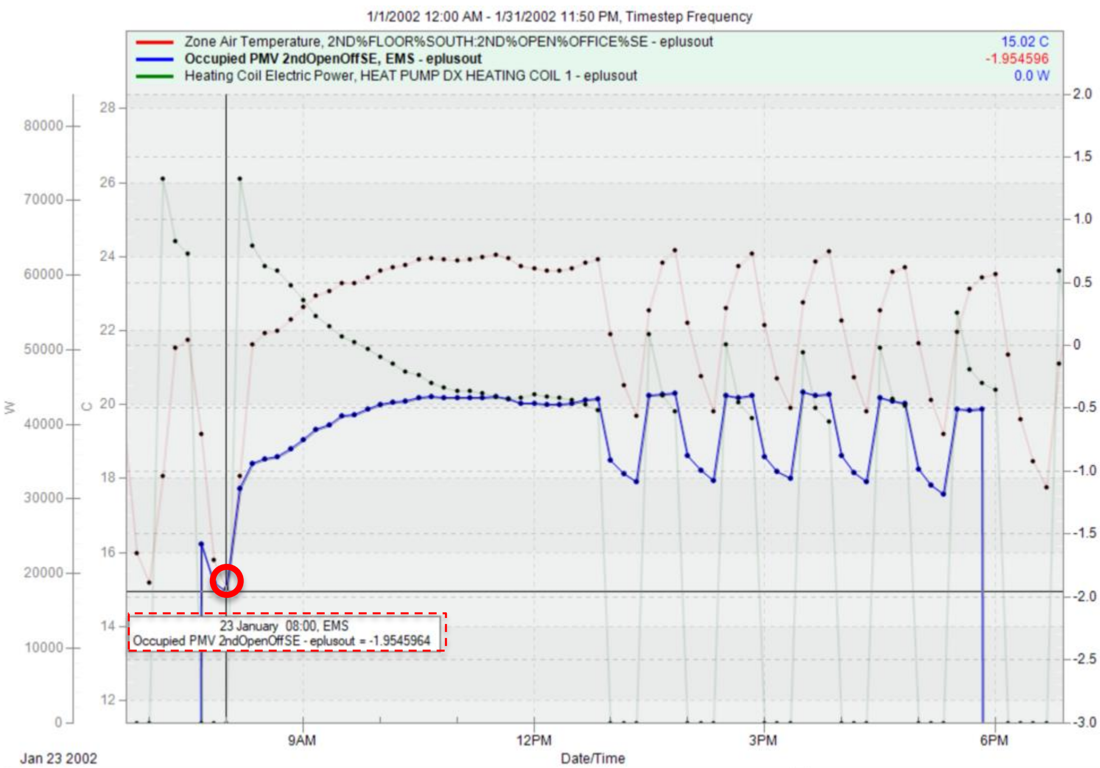


Figure 2.1.8 Lowest PMV value of the simulation time period

The highest value for PMV is +0.8 at the open office area on Southeast on the first floor, at the time of 3:30 pm on Jan 13th, during occupied hours, as indicated in Figure 2.1.9 – 2.1.10. At that moment, the indoor air temperature reached around 27.5 °C, without any cooling system for this scenario, it's hard to change the PMV when it's warm. The heat pump is turned off during that day, and outdoor dry-bulb temperature also reached the peak during the simulation period, as shown in Figure 2.1.2 and Figure 2.1.12.

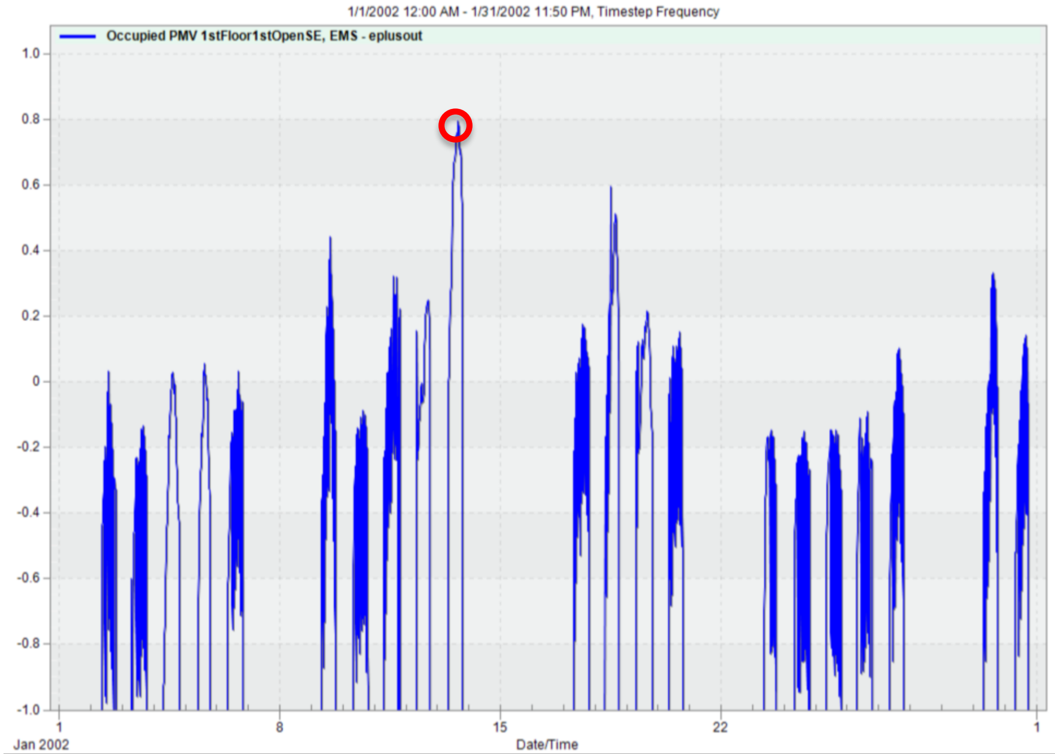


Figure 2.1.9 PMV for the open office on the Southeast on 1st floor

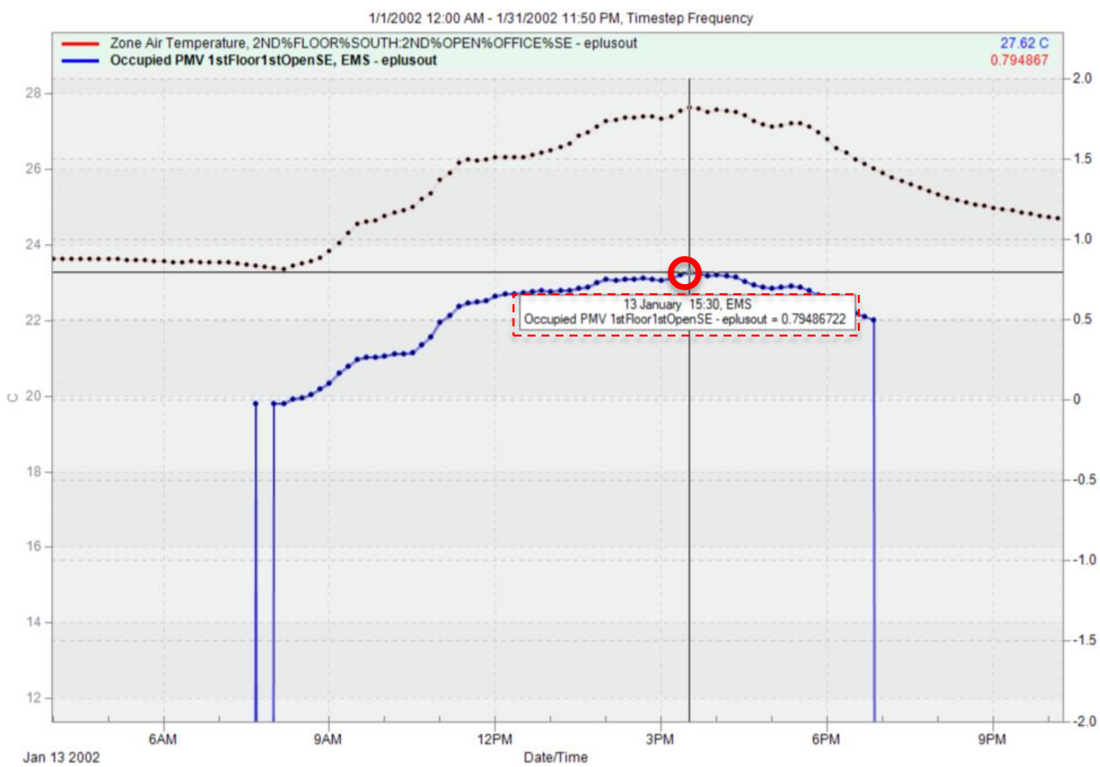


Figure 2.1.10 Highest PMV during simulation time period

Comfort Percentage

In Rstudio, the result data is divided into subset for each zone, with the data of Outdoor dry-bulb temperature, outdoor relative humidity, indoor air temperature, setpoint temperature, HVAC electric demand, and heating coil power. A new column is added as indicator for Comfort. Based on ASHRAE standard 55 (2010), the acceptable range for PMV as comfortable is from -0.5 to 0.5. When the PMV value falls in the acceptable range, the value for Comfort is “1”, otherwise is “0”. A R code example is here:

```
#create new column indicating if the PMV is in the range of comfort
wknd_occupied <- wknd_occupied %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))
```

The mean of the dichotomous variable “Comfort” is the percentage of the comfort. Then, the comfort percentage was calculated for each zone. The classroom on the first floor is occupied during weekends, all the rest of the zones are occupied during weekdays, so their effective PMV values are at different timestep, therefore the average comfort percentage were calculated as two types: weekdays and weekends. The average comfort percentage during weekday is 67.43%, which during weekend is 45%.

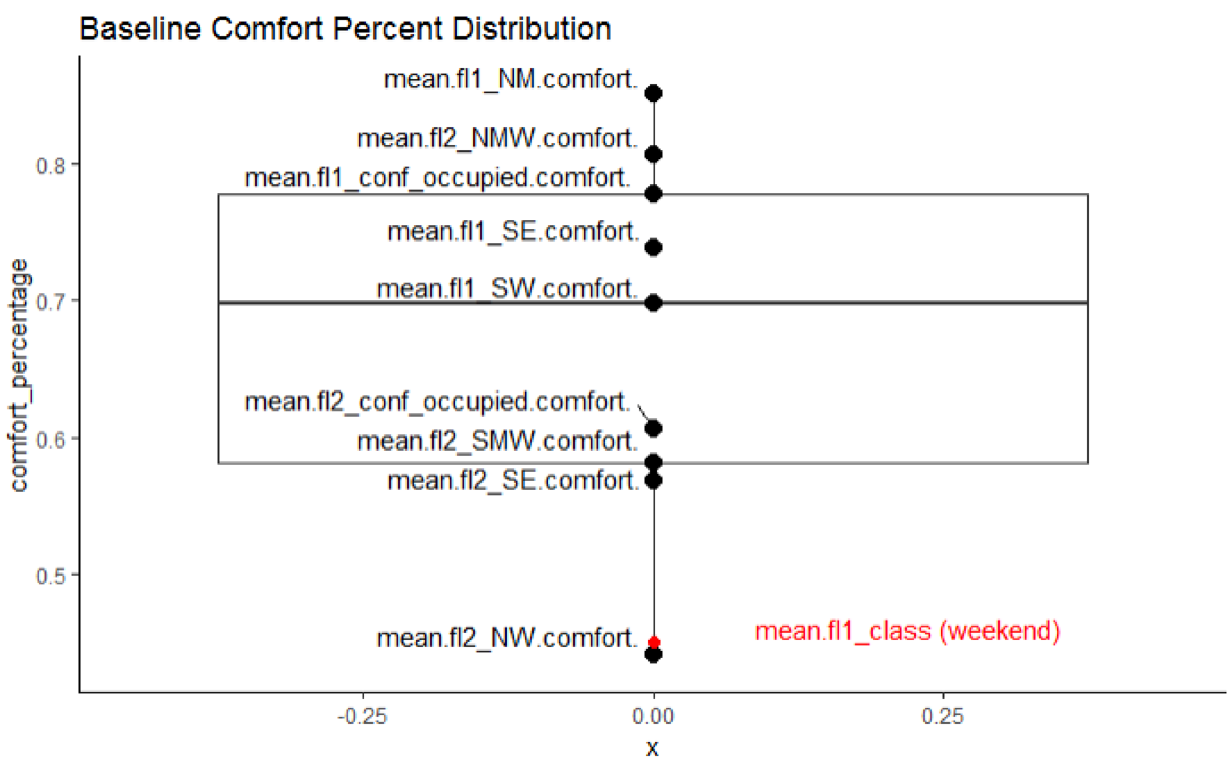


Figure 2.1.11 Baseline Comfort Percentage Boxplot

Heating Electric Power

The heat pump on or off condition can be identified by the real time heating coil power data. When it's 0, it means the heat pump is off.

Table 2.1.4 HVAC Electric Power

	HVAC_elec_demand (w)	heating_coil_power (w)
Min	42529	0
1st Q	42529	0
Median	42529	0
Mean	61247	18718
3rd Q	86998	44469
Max	**122790**	80261

As indicated in Figure 2.1.12, there are many times the heat pump is on even no one occupied the space, and it's at the maximum power rate. This is a potential opportunity for reducing the HVAC energy consumption.

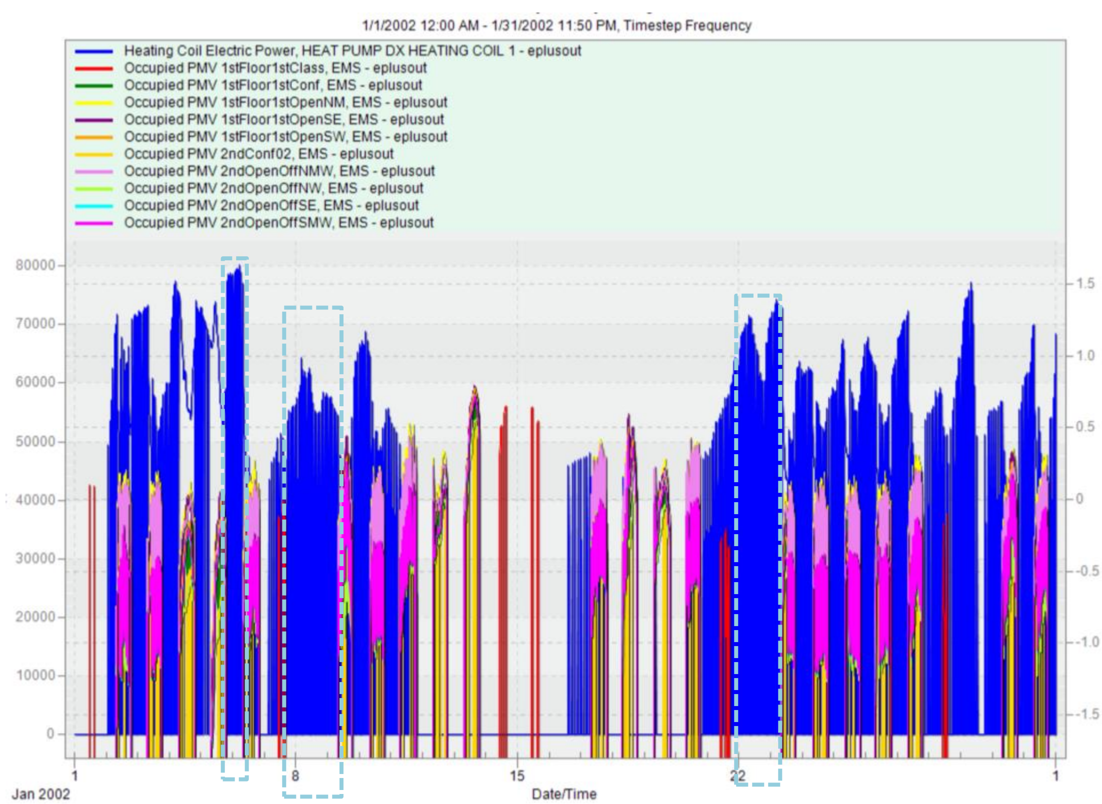


Figure 2.1.12 Heating Coil Electric Power VS Effective PMV

The total heating energy consumption in calculated in Rstudio based on the following equation:

```
#total heating energy consumption
heating_energy <- mean(baseline$heating_coil_power) * 31 *24 / 1000
heating_energy

## [1] 13926.26
```

The unit of heating coil power is in watt, so the average heating coil power multiply the days and hours and divided by 1000, the result is the total heating energy consumption in kWh, which is 13926.26 kWh for baseline heating energy consumption.

2.2. Control Problem Formulation

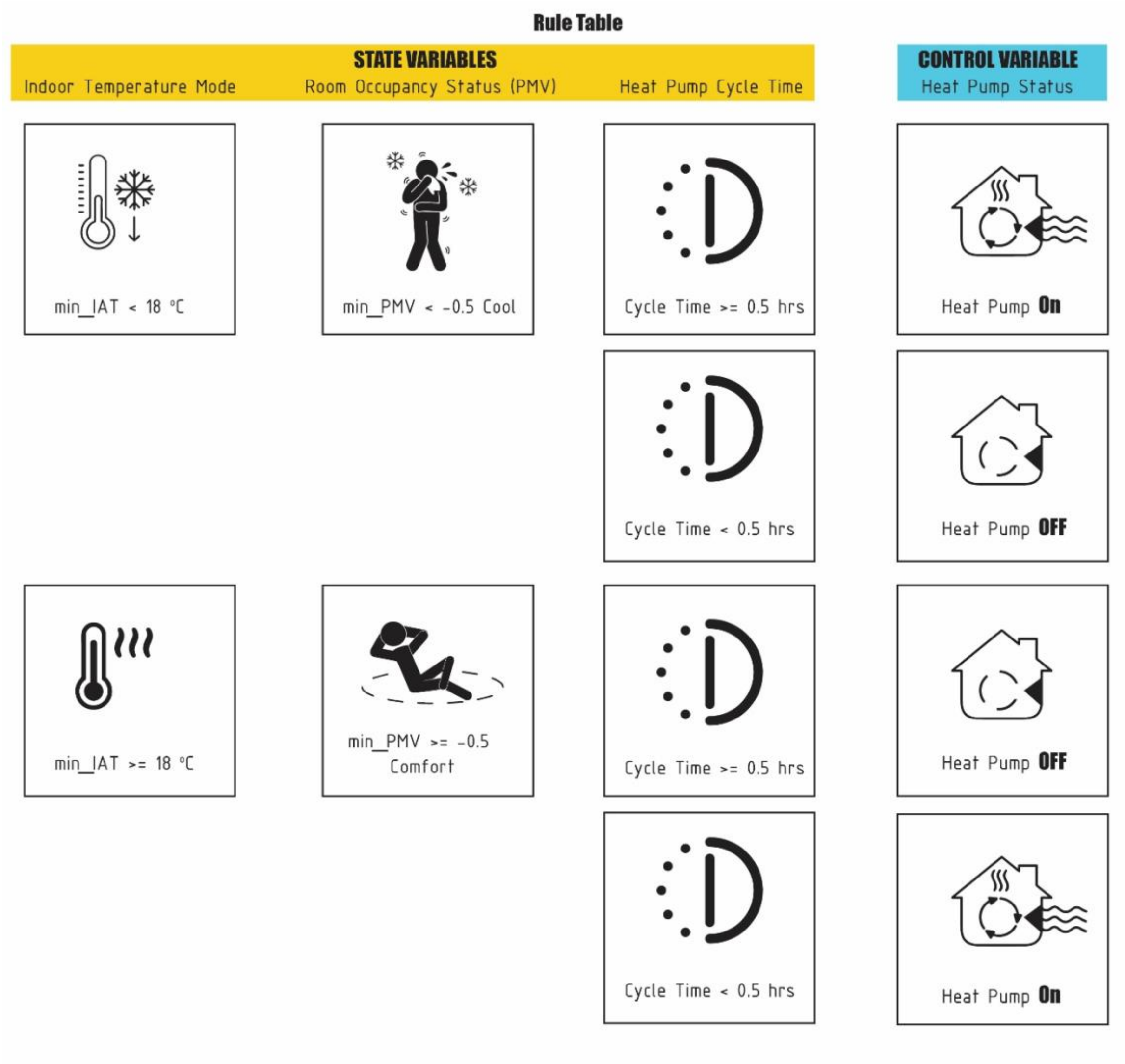


Figure 2.2.1 Baseline Control Rule Table

State variables

The state variables in the baseline control logic are the minimum indoor air temperature among all zones, and the minimum effective PMV among all zones. In this scenario, one more state variable is the heat pump cycle time, which cannot be changed to be less in the proposed control strategy. Other possible state variables to test for proposed control logic are average effective PMV among all zones, outdoor condition, occupancy status, etc.

Control variables

The control variable in this case is just the heat pump status, to turn on or off the heat pump.

Control Objectives

There are two objectives of the proposed control strategy:

- To reduce HVAC energy consumption
- To improve comfort percentage

In the Rstudio analysis, the total heating energy consumption and average comfort percentage are identified as evaluation indicators. The same methodology is used to calculate the evaluation results for proposed control logics.

3. Control Logic Design

As discussed in the previous section, the control objective in the study was to minimize the HVAC energy consumption and to improve comfort conditions for the occupants. To do so, the state variables in the baseline control logic were analyzed. The control variables are simple on/off controls, without any variation in percentage of output. Since the output percentage can't be varied in the given system, our approach was to arrive at an optimum time to switch the system on and off in order to achieve our objectives. The short cycling limitation over the control variables was also constant over all the explored scenarios. A description of the different strategies explored in the study is provided in this section. The modifications were made sequentially, to check the effectiveness of modifying the control points for different state variables. Control Scenarios 1, 2 and 3 describe the different modifications made to the baseline control logic.

3.1. Control Scenario 1 - Occupancy Based Control

Since the HVAC system has a single control to switch on or off for the whole building, there is no possibility of turning it off based on unoccupied zones. Hence, our approach was to identify times when the building was completely unoccupied. In the baseline code, the temperature constraint (not letting the IAT drop below 18 °C) keeps the system on even during the nights and on the weekends, when the building is more or less unoccupied.

The baseline control logic checked the entire list of PMV values, and utilized the minimum value to check if the system needed to be switched on or kept on. Whenever there was no occupancy in a particular area, the value "-9999" was returned. The list was then parsed to filter out the observations with a value of -9999, and a list of effective PMVs was created. In our modified control logic, the variable containing all real PMV values (effective PMV values) was checked. If the length of this list was zero, the building was interpreted as unoccupied. Figure 3.1.1 illustrates the code used to do this.

```
if cycle_count >= 3:
    # Turn on heating if the conditions allow
    if min_pmv < -0.5 or (min_iat < 18 and len(eff_pmvs) != 0):
        act = 1;
    else:
        act = 0;
if act == dx_status:
    # Remember cycle number
    cycle_count += 1;
else:
    cycle_count = 1;
```

Figure 3.1.1 Control Logic to turn off system when the building is unoccupied

The simulation was then run for this updated control logic. As a result of this modification, the energy consumption decreased from 13926 kWh to 8958.69 kWh. However, the percentage of comfort decreased from 67.4% to 46.77%. Since one of our control objectives was to increase comfort, this result was not favorable. This was possible because of the large temperature differential caused by completely switching off the heating system in peak winter. Even aside from the energy consumption, it is not advisable to completely switch off temperature control in buildings to prevent the water in pipes from freezing, and to protect other equipment and systems in buildings.

3.2. Control Scenario 2 - Set Point and Set Back Temperature Control

Since the system switching off was not effective, our next approach was to create a scheduled set-back temperature for the night time, since the building was always occupied during the night time. The Indoor Air Temperature (IAT) state variable was used as a control point to switch the system on or off. The code used to implement this logic is illustrated in Figure 3.2.1.

```
if cycle_count >= 3:
    #during the night
    if time > 19 or time < 5:
        #turn on heating
        if min_iat < 12:
            act = 1;
        else:
            act = 0;

    #during the day, turn on heating
    else:
        if min_pmv < -0.5 or min_iat < 18:
            act = 1;
        else:
            act = 0;
```

Figure 3.2.1 Control Logic to implement a scheduled set-back temperature during the night time.

When this logic was implemented using EnergyPlus, the consumption was found to be 11,606 kWh, which was higher than the previous scenario, but still lower than the baseline condition. However, the occupant comfort rate during the weekdays was still found to be lower than the baseline condition, at 60.4%.

3.3. Control Scenario 3 - Set point and Set Back Control + Refined PMV Constraints

To achieve the objective of improving comfort conditions in the building, the process of applying constraints on PMV were studied. The baseline logic used the minimum PMV value as a constraint to switch the system on and off. However, a better representation could be to take into account the mean value of PMV for the different zones, since the system is common for all the locations. In case there is one zone with an extremely low PMV value while all others are satisfied, there wouldn’t be an unnecessary increase in temperature. To improve the comfort values, the treshold for PMV was increased from -0.5 to 0. As a result, this control logic was based on adding a constraint of having mean PMV above 0, while also implementing scheduled setpoint and setback temperatures. The rule table illustrated in Figure 3.3.1 represents the logic.

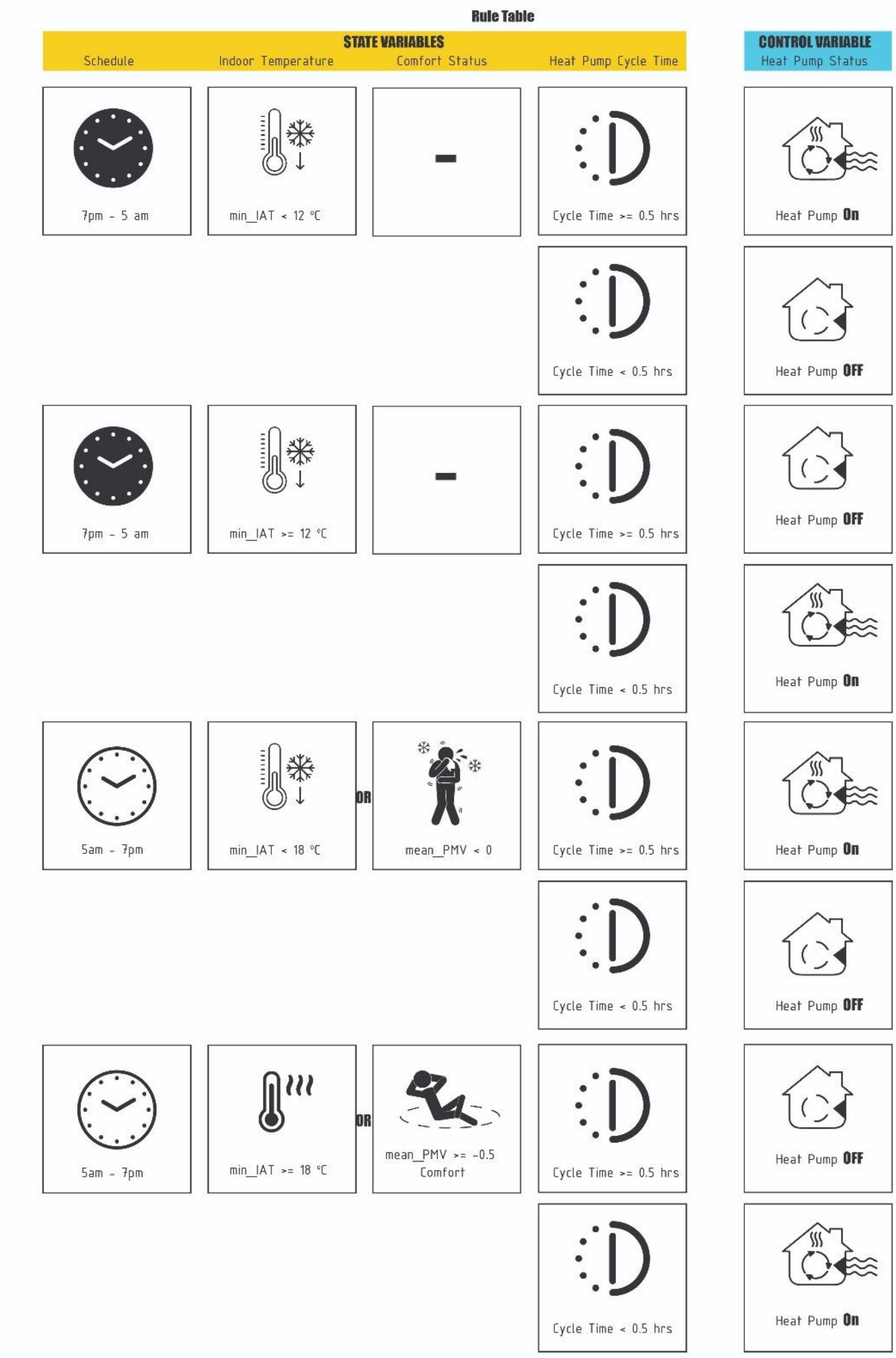


Figure 3.3.1 Control Scenario 3 - Rule Table

The code used to implement this logic, and the control diagram for this scenario are described in detail in Section 4. As a result of implementing this logic, the heating energy consumption was found to be 12,365 kWh (which is a reduction compared to the baseline consumption of 13,926 kWh). The comfort percentage also increased to 75.38%. Both our control objectives were achieved using Control Scenario 3 – a combination of setback temperature and PMV based control.

3.4. Summary of Control Logic Selection

A summary of the heating energy and comfort percentage results for the baseline control scenario, and control scenarios 1,2 and 3 is given in Table 3.4.1. The variation in comfort percentage in the different control scenarios is shown in Figure 3.4.1.

Table 3.4.1 Summary of Heating Energy Consumption, Weekday Comfort Percentage and Weekend Comfort Percentage

	Heating Consumption (kWh)	**Weekday Comfort %**	**Weekend Comfort %**
Baseline Control	13926.26	0.6743329	0.45
Control Scenario 1	8958.69	0.4670576	0.36
Control Scenario 2	11606.92	0.6046910	0.47
Control Scenario 3	12365.91	0.7538564	0.64

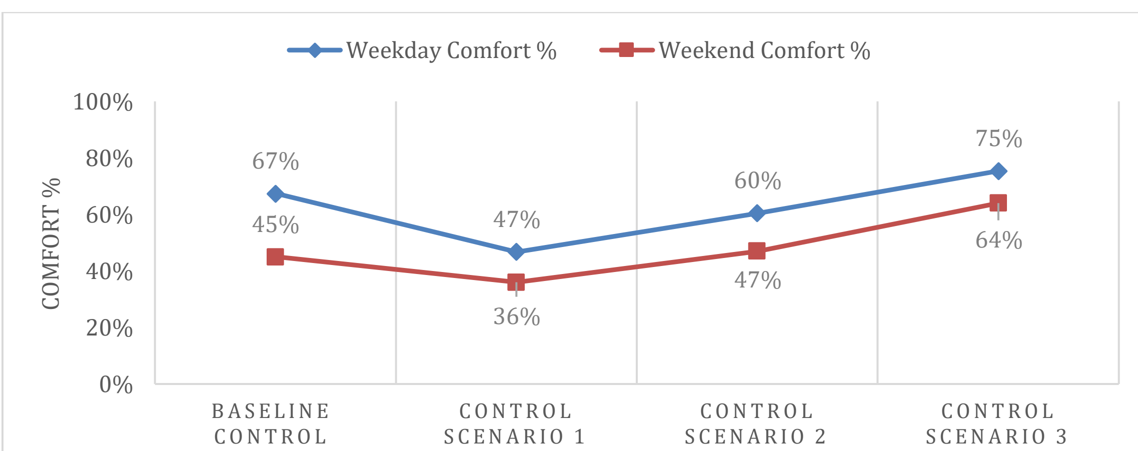


Figure 3.4.1 Weekend and weekday comfort percentage for different control scenarios

The change in heating energy consumption values is shown in Figure 3.4.2. From these observations, we can conclude that Scenario 3 is the most appropriate control logic to achieve this study's control objectives.

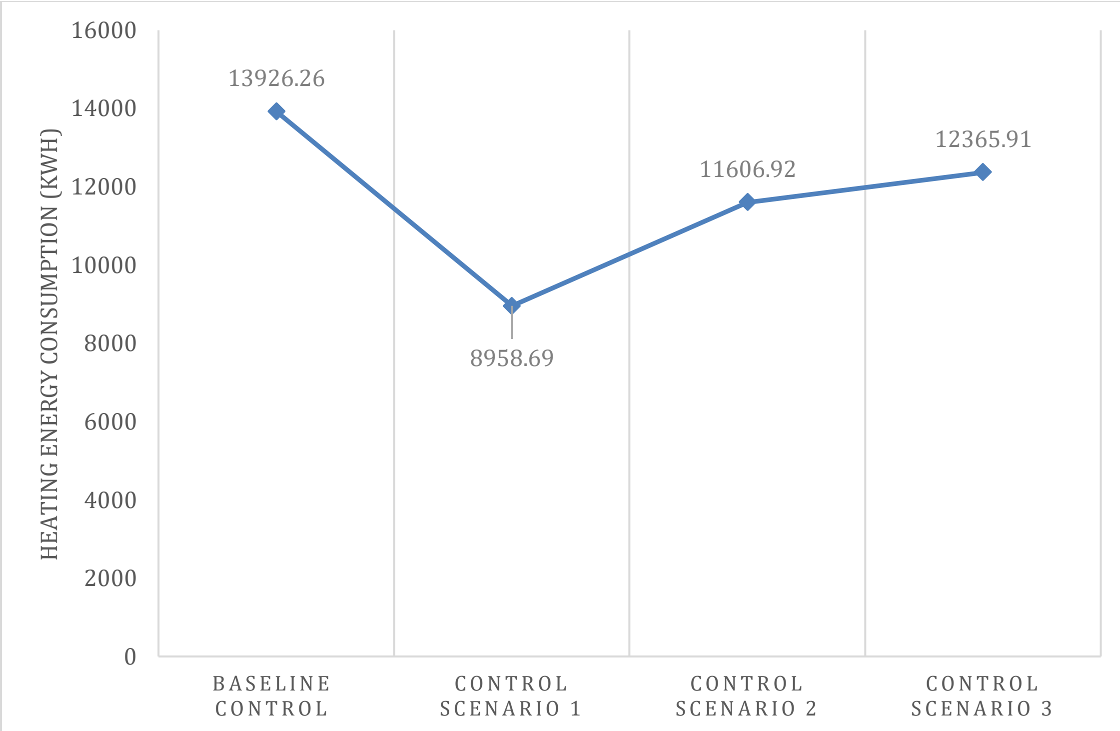


Figure 3.4.2 Heating Energy Consumption in the different control scenarios

4. Control Logic Implementation and Simulation

Based on the analysis above, it is reasonable to add a HVAC night setback in control logic to save energy. In addition, the condition of minimum PMV only considers the most dissatisfied person, so it is better to use mean PMV. In this way, most of the occupants' thermal comfort would be considered. A new variable is set to store time information, and a new condition logic is set to check if the time is during the daytime. If the variable is less than 5 or larger than 19, then it is during the night, and heat pump's objective would be keeping the indoor air temperature at 12˚C. Otherwise, it is during the daytime, then heat pump's objective is to not only keep all zone's air temperature above 18˚C, but also make the average thermal comfort level not less than 0. This new optimized the control logic is represented in a new flow chart, as indicated in Figure 4.1. Compared with baseline flow chart (Figure 2.1.1), it is clear that the only change is time check condition and temperature setpoint condition during the night.

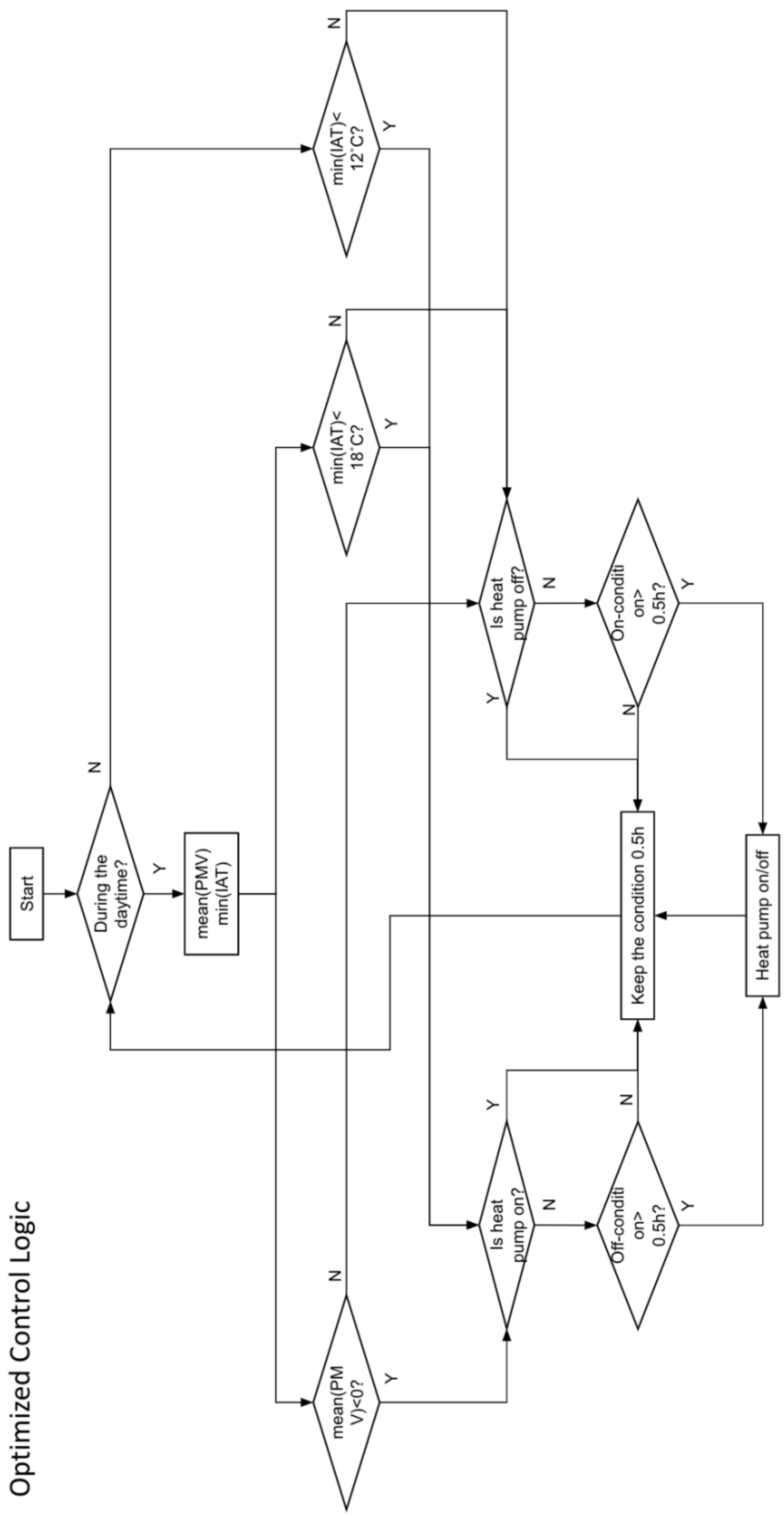


Figure 4.2 Optimized Control Logic Flow Chart

After creating a virtual environment in terminal and entering it, a client script file (xremote.py) is used to send the new code with optimized control logic to a remote server in Intelligent Workplace, and then simulation results are sent from the server. The whole process is conducted with EnergyPlus-based building control test-bed.

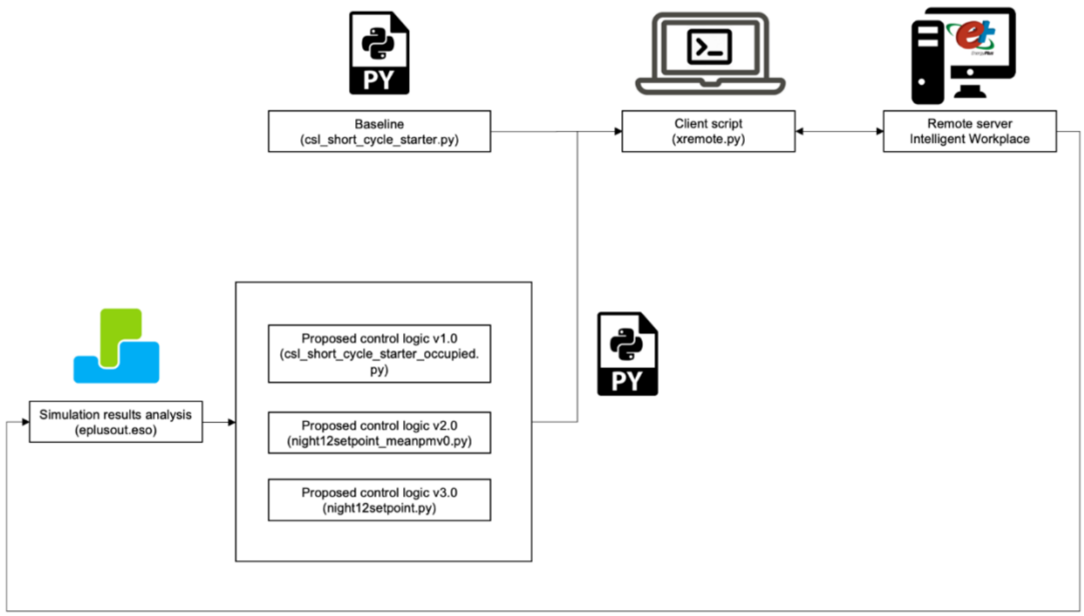


Figure 4.3 EnergyPlus-based building control test-bed framework

5. Comparative Evaluation

5.1. Indoor thermal comfort performance – PMV

Generally, when occupants feel dissatisfied about the thermal comfort, PMV will either rise above 0.5 or drop below -0.5. Therefore, we have to divide comparative evaluation between baseline and alternatives into 2 parts (above 0.5 and below -0.5).

5.1.1. Baseline Condition

For open office and conference room, the percentage of PMV within comfort range is 67.4% and for classroom, the percentage of PMV within comfort range is 45%. The percentage of PMV within comfort range in each zone is shown in figure 5.1-1. Detailed information about PMV in each zone is summarized in Table 5.1-1.

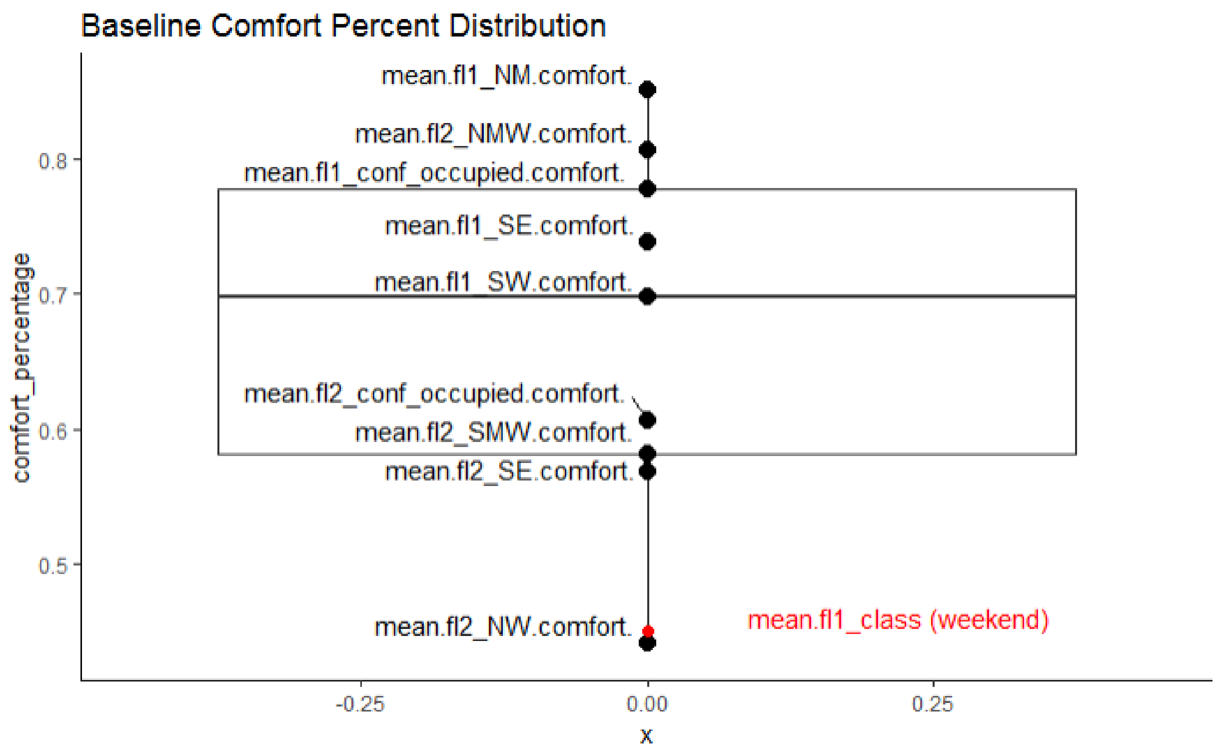


Figure 5.1-1 Baseline Comfort Percent Distribution

Table 5.1-1 Summaries of information about PMV in each zone

PMV	1_conf	1_class	1_NM	1_SW	1_SE	2_conf	2_SMW	2_SE	2_NMW	2_NW
Min	-0.967	-1.089	-1.4099	-1.7716	-1.7012	-1.176	-1.9408	-1.9546	-1.4432	**-1.9173**
1st Q	-0.421	-0.567	-0.2844	-0.5307	-0.4644	-0.626	-0.6577	-0.6707	-0.3631	**-0.7866**
Median	-0.202	-0.118	-0.0190	-0.2968	-0.2211	-0.425	-0.4225	-0.4336	-0.0763	**-0.5205**
Mean	-0.229	-0.077	-0.0761	-0.3063	-0.2374	-0.444	-0.4280	-0.4385	-0.1368	**-0.5422**

3rd Q	-0.063	0.518	0.1251	-0.0760	-0.0062	-0.278	-0.1945	-0.2025	0.0685	**-0.3153**
Max	**0.741**	**0.654**	**0.7928**	**0.7742**	**0.7949**	**0.564**	**0.7209**	**0.7207**	**0.7357**	**0.5541**

When PMV rises above 0.5, occupants feel hot due to the rise of indoor air temperature and start to be dissatisfied with the thermal comfort. However, the maximum PMV (0.7949) occurs at the southeastern part of the open office on the first floor and it exceeds the upper bound (0.5) by 0.2949, which means occupants will experience low level of discomfort but not much.

Major time period when PMV for open office exceeds 0.5 falls between Jan 13th and Jan 15th. Only for a few certain time points on Jan 11th and Jan 18th, PMV for open office area on the first floor exceeds 0.5 by 0.04 at most. Thus, we focus on the analysis of major time periods, especially for Jan 13th.

As shown in figure 5.1-2, for open office area, except the northwestern area on the second floor, the discomfort time period generally starts between 11.00 and 11.30 on Jan 13th, and it ends between 18.30 and 18.50 on Jan 13th. Occupants there will feel slightly uncomfortable for about 7 to 8 hours, almost one work day. For the northwestern area on the second floor, the discomfort time period starts at 14.40 and ends at 17.30, lasting for 2 hours and 50 minutes.

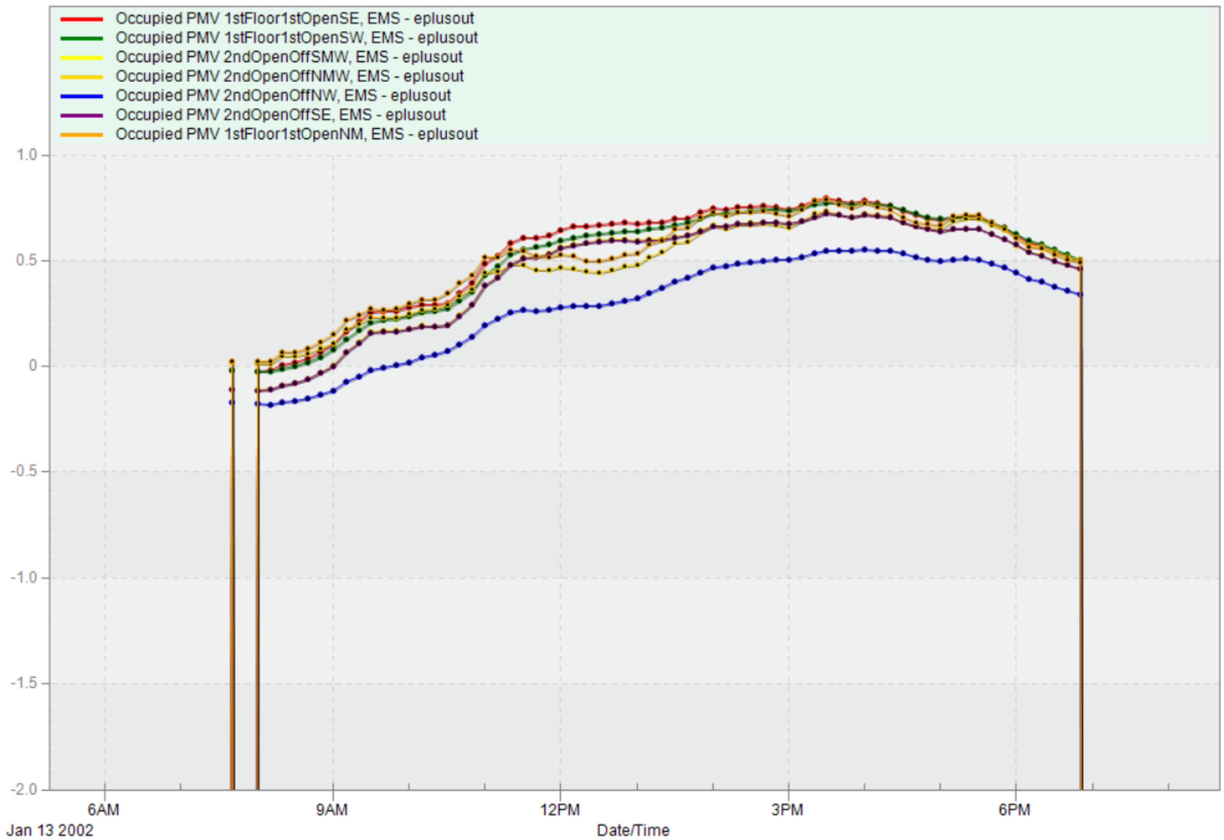


Figure 5.1-2 PMV fluctuation for open office on Jan 13th

As shown in figure 5.1-3, for conference room on the first floor, PMV exceeds 0.5 during the following time period on Jan 13th:

- From slightly later than 14.00 to 15.00
- From 16.00 to 17.00

PMV is slightly lower than 0.5 at 14.00 (the beginning of the first conference) and it rises above 0.5 in a few minutes. For conference room on the second floor, PMV exceed 0.5 from slightly later than 16.00 to 17.00. Overall, for both conference rooms, so long as occupants there start to feel uncomfortable, the period will last for about 1 hour or exactly 1 hour.

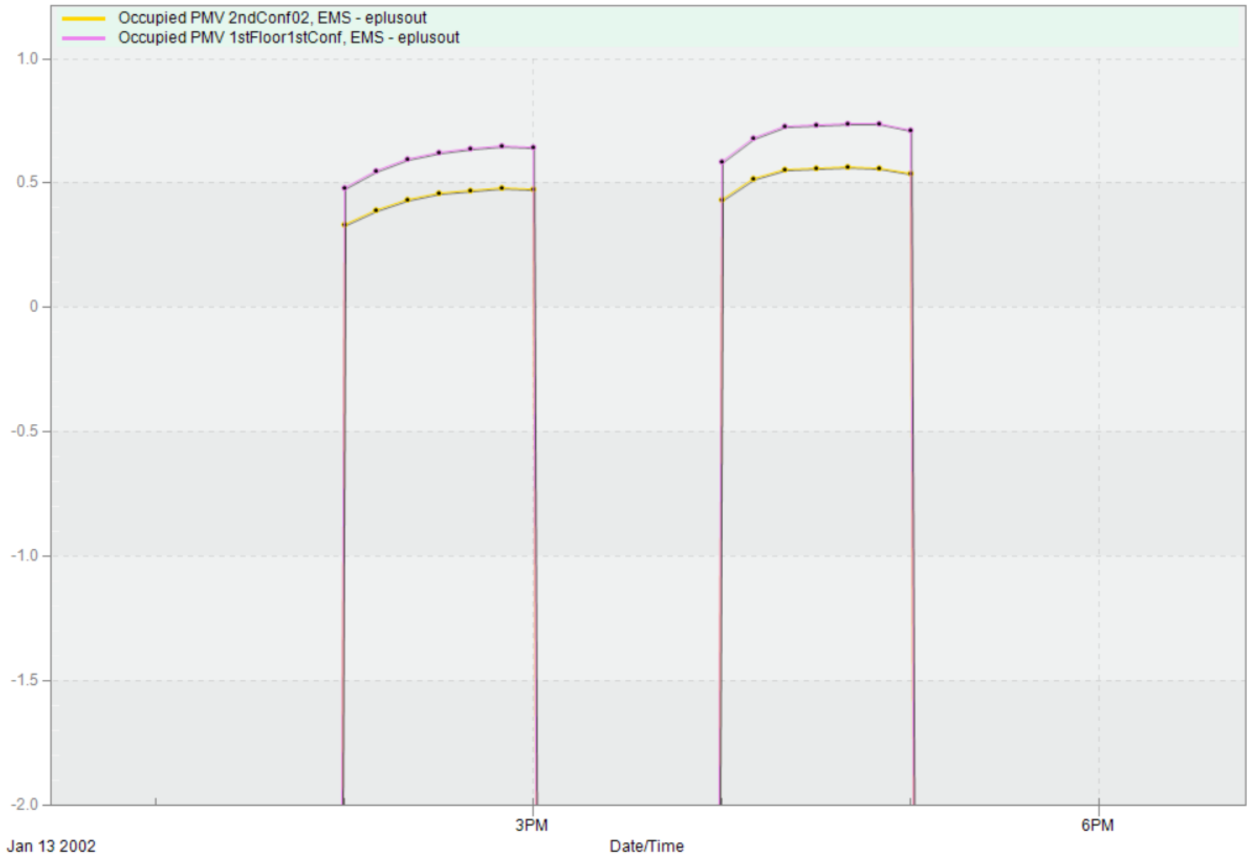


Figure 5.1-3 PMV fluctuation for conference room on Jan 13th

As shown in figure 5.1-4, PMV for classroom exceeds 0.5 during the following period:

- From 11.20 to 11.30 on Jan 14th, near the end of the class
- From slightly later than 13.30 to 15.30 on Jan 14th, almost the whole class
- From 10.30 to 11.30 on Jan 15th, the whole class
- From15.00 to 16.00 on Jan 15th, the whole class

Generally, when occupants there will feel uncomfortable during the whole class, up to about 2 hours, except for the first class on Jan 14th.

In this case, heat pump only provides heating, thus when PMV exceeds 0.5, the most we can do is to switch off heat pump. And during the periods when PMV exceeds 0.5, heat pump is already switched off, thus we cannot solve thermal discomfort issues incurred by high outdoor air temperature. Therefore, we will focus on solving the thermal discomfort issues incurred by low outdoor air temperature.

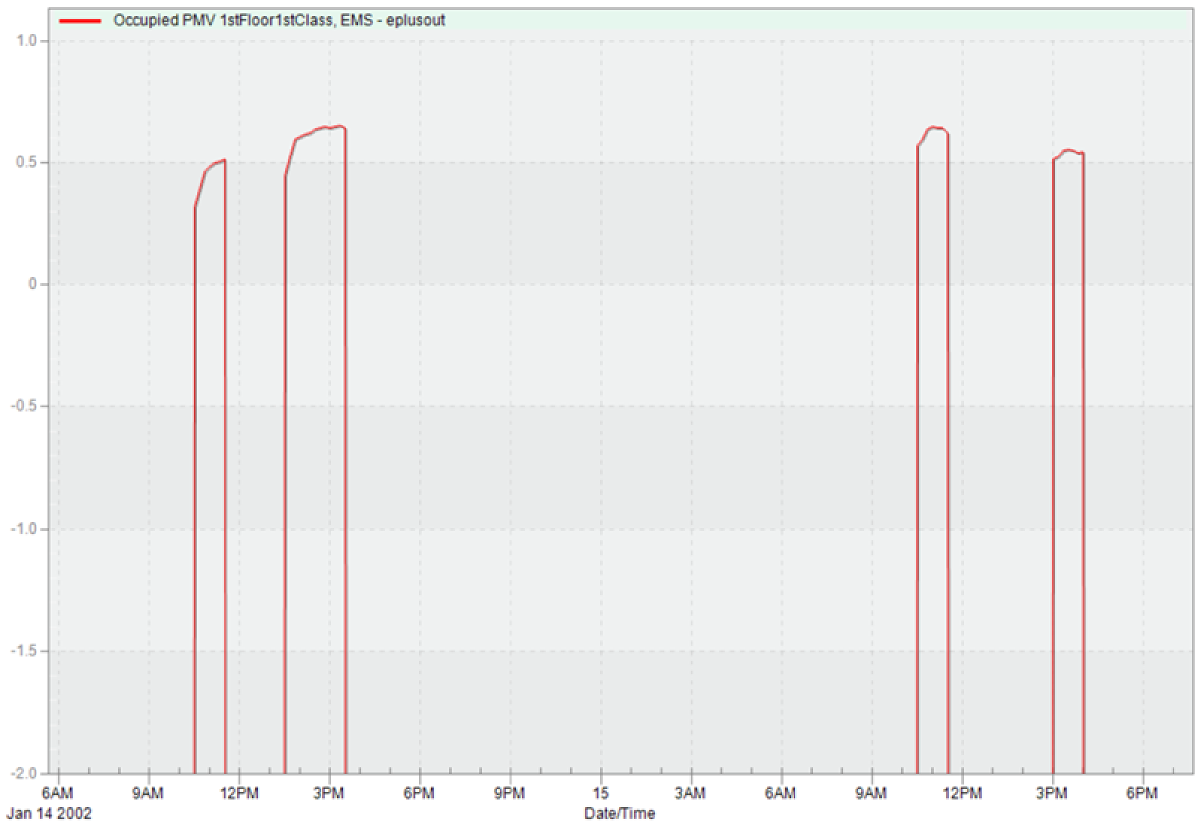


Figure 5.1-4 PMV fluctuation for classroom on Jan 13th

When PMV drops below -0.5, occupants feel cold due to the drop of indoor air temperature and start to be dissatisfied with the thermal comfort. The minimum PMV (-1.9546) occurs at the southeastern part of the open office on the second floor and it is -1.4546 less than the lower bound (-0.5), which means occupants can experience relatively high level of discomfort due to the low temperature.

As shown in figure 5.1-5, for open office area, typical PMV pattern when occupants experience discomfort is like sawtooth. PMV fluctuates around -0.5, and occupants there experience about 30 minutes of comfort and about 30 minutes of discomfort alternatively.

However, when outdoor air temperature is under -10°C, another two patterns occur in the open office of the second floor (expect zone NMW). As shown in figure 5.1-6, pattern 1 is like sawtooth as well, but the upper bound of sawtooth is around -0.5, which means when PMV reaches -0.5 for a few minutes, heat pump is switched off and consequently indoor air temperature and PMV drop for 30 minutes until heat pump can be switched on. In this pattern, for most of the time, occupants feel cold and uncomfortable. As shown in figure 5.1-7, pattern 2 is like mountain, and PMV cannot reach -0.5 and occupants always feel cold and uncomfortable. The reason is that outdoor air temperature is below -15°C and that heat pump cannot provide enough heating to rise PMV of these zones to -0.5.

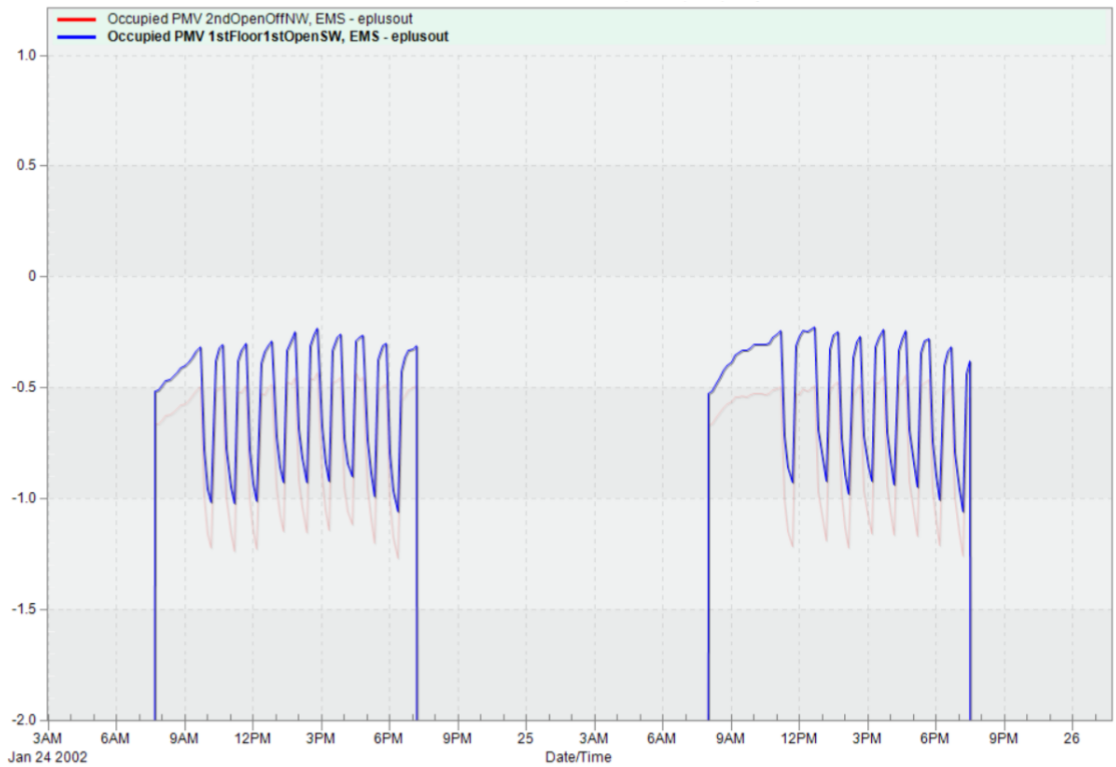


Figure 5.1-5 Typical pattern of PMV when occupants feel uncomfortable in open office

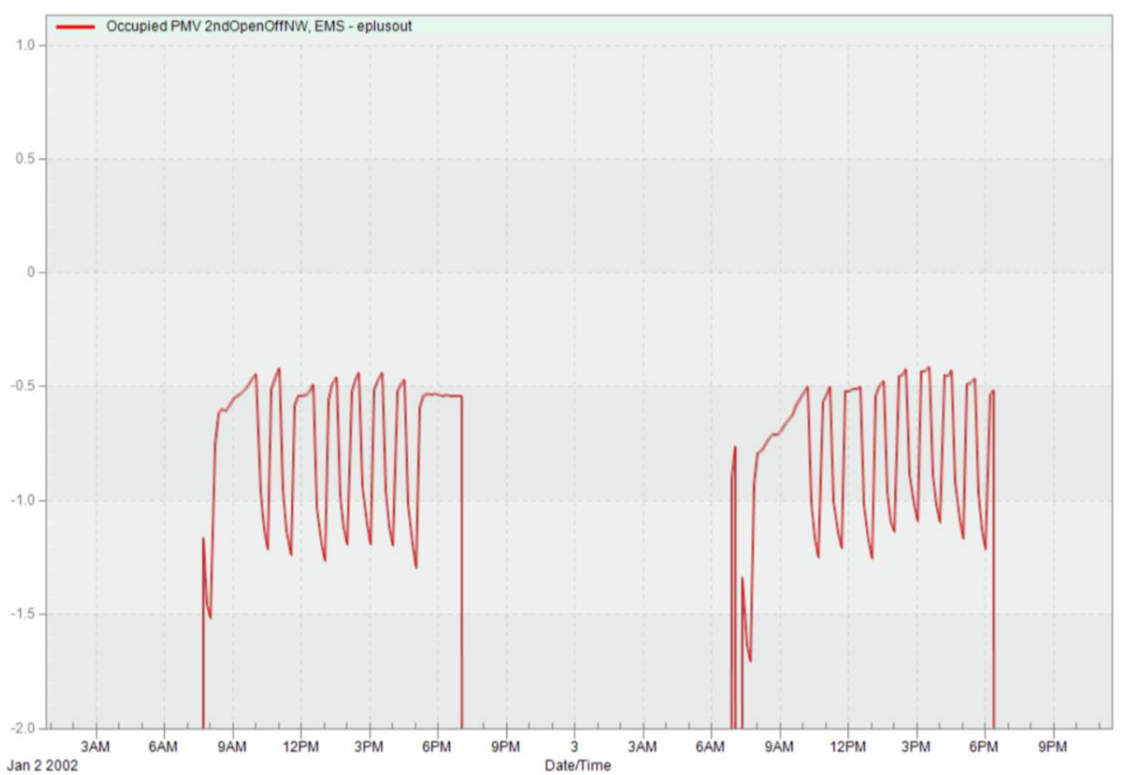


Figure 5.1-6 PMV pattern 1 when outdoor air temperature is under -10°C

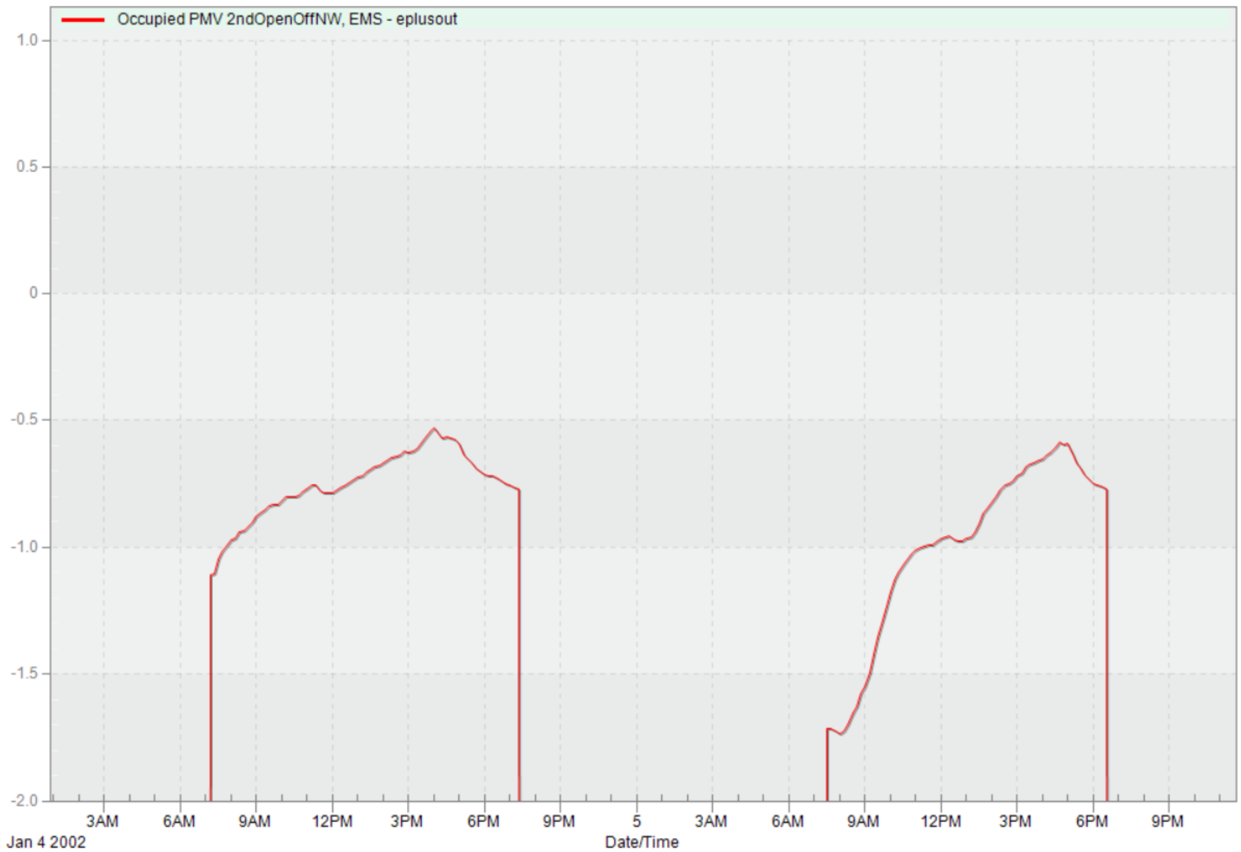


Figure 5.1-7 PMV pattern 2 when outdoor air temperature is under -10°C

As shown in figure 5.1-8, typical pattern of PMV in the conference room when occupants experience discomfort is similar to that in the open office: PMV fluctuates around -0.5 and occupants there experience about 30 minutes of comfort and about 30 minutes of discomfort alternatively.

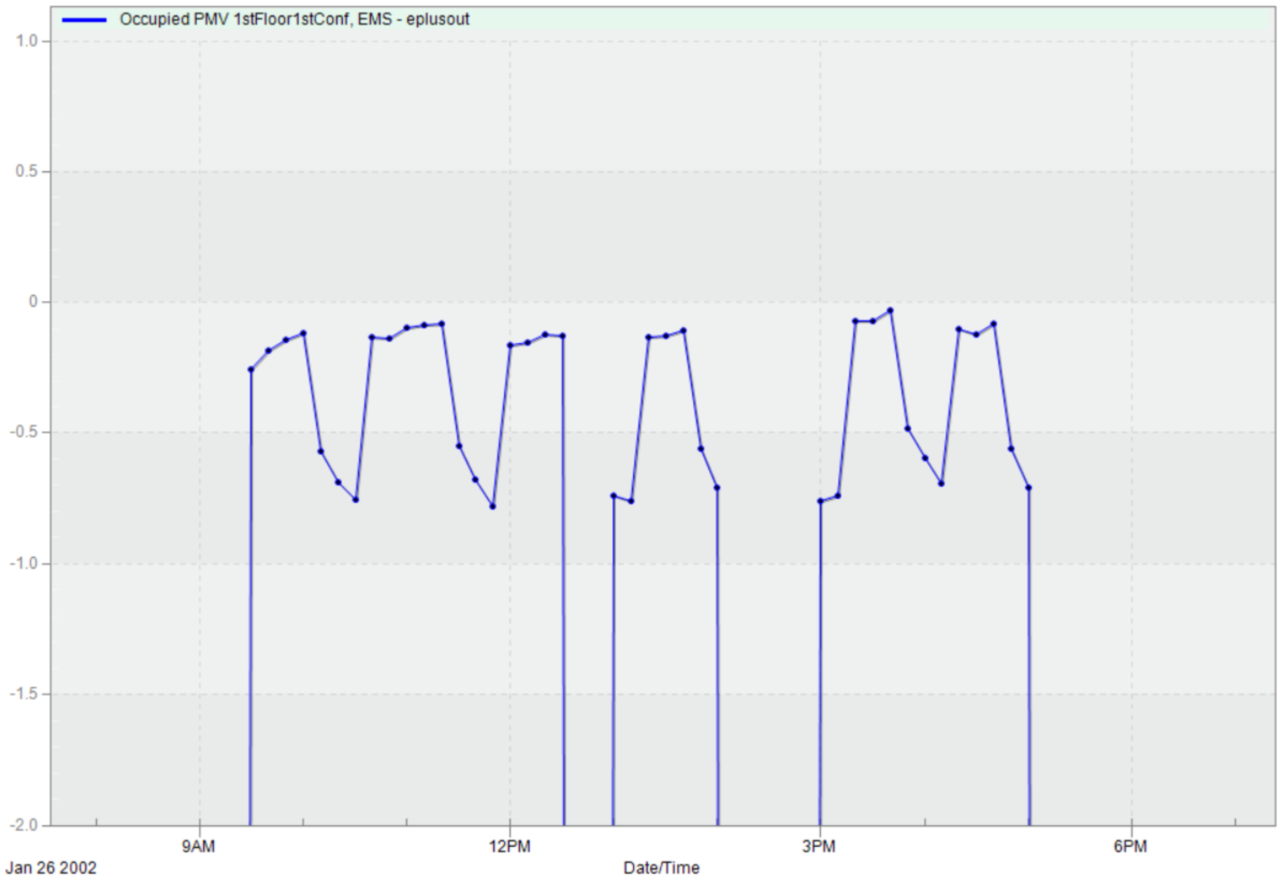


Figure 5.1-8 Typical pattern of PMV when occupants feel uncomfortable in conference room

As shown in Figure 5.1-9, 5.1-10 and 5.1-11, PMV in classroom fluctuate around -0.5 and occupants there experience about 30 minutes of comfort and about 30 minutes of discomfort alternatively.

Figure 5.1-9 PMV fluctuation in classroom on Jan 7th

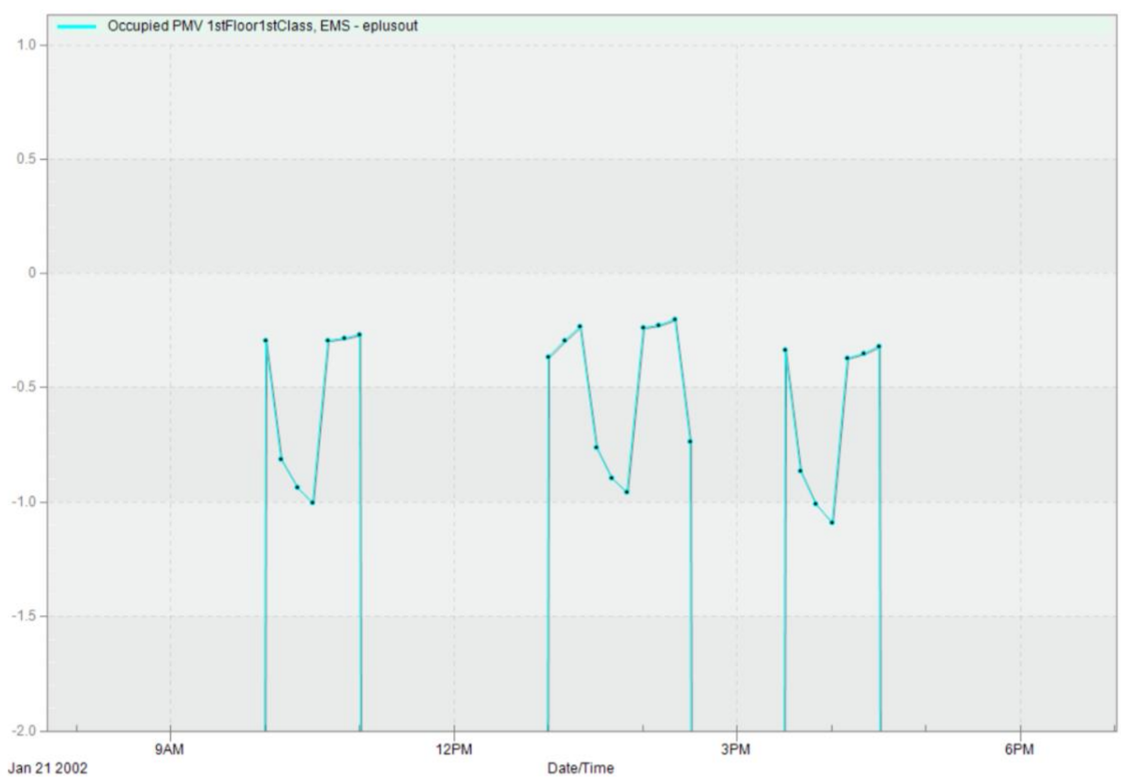


Figure 5.1-10 PMV fluctuation in classroom on Jan 21st

Figure 5.1-11 PMV fluctuation in classroom on Jan 28th

5.1.2. Switch off heat pump or Lower Setpoint to 12°C at night

In order to reduce energy consumption of heat pump, we tried two scenarios:

- Switch off heat pump at night
- Lower setpoint temperature to 12°C at night

These two can reduce energy consumption, but they also compromise thermal comfort as well. If we switch off heat pump at night, the percentage of PMV within comfort range drops from 67.4% to 46.7% in open office and conference room and the percentage of PMV within comfort range drops from 45% to 36% in classroom. In this scenario, the percentage of PMV within comfort range in each zone is shown in figure 5.1-12.

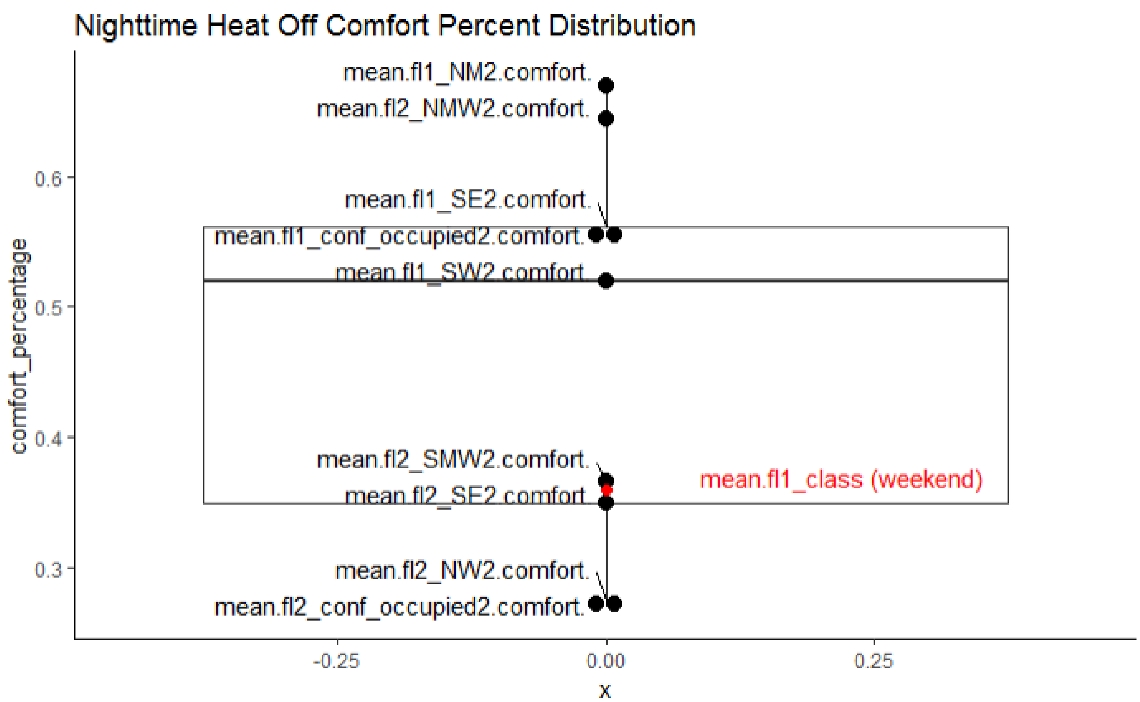


Figure 5.1-12 Nighttime Heat off Comfort Percent Distribution

If we lower setpoint temperature to 12°C at night, the percentage of PMV within comfort range drops from 67.4% to 60.5% in open office and conference room and the percentage of PMV within comfort range rises from 45% to 47% in classroom. In this scenario, the percentage of PMV within comfort range in each zone is shown in figure 5.1-13.

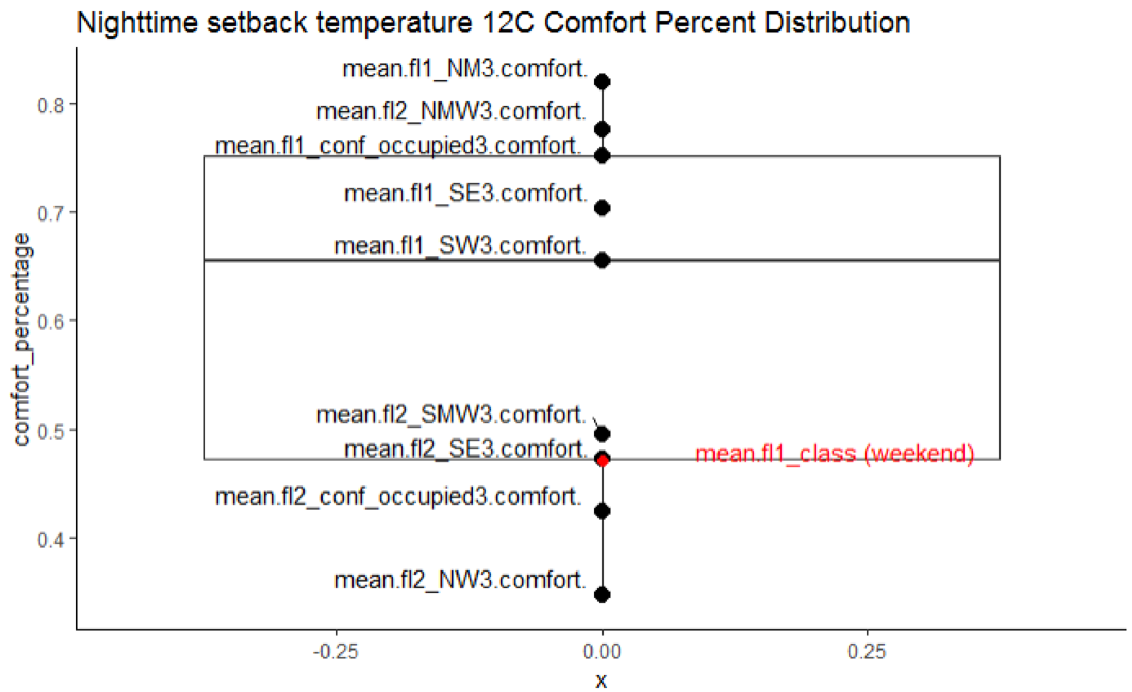


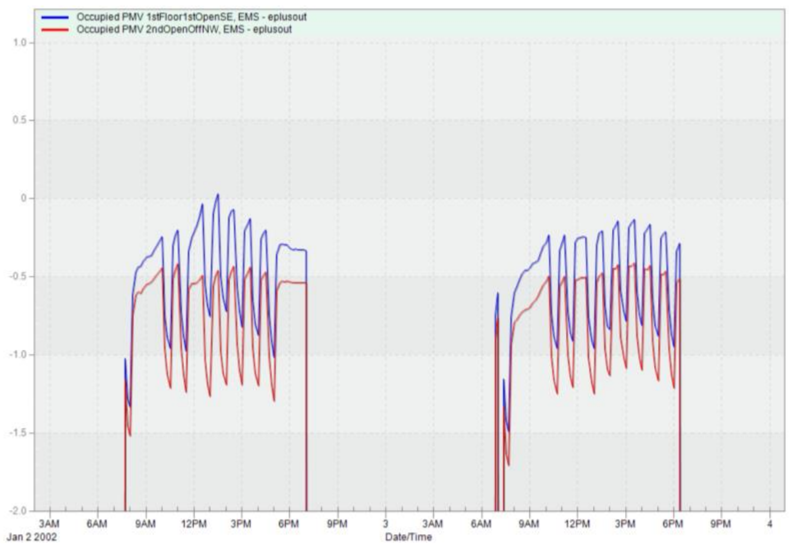
Figure 5.1-13 Nighttime Setback Temperature 12°C Comfort Distribution

As indicated in table 5.1-2, in the open office area, it takes just different time for three scenarios to reach the control target (-0.5), with switching off heat pump consuming most time, and then the pattern of PMV is similar. However, when outdoor air temperature is extremely low, for both alternatives, PMV cannot rise up to the control target (-0.5) due to the capacity limit, just like baseline case. Generally, when outdoor air temperature is normal, thermal comfort in the early morning is compromised due to setting lower setpoint temperature or switching off the heat pump at night.

As indicated in table 5.1-3, in the conference room, both alternatives consume more time to raise PMV to the control target (-0.5) than baseline does, with switching off heat pump consuming most time. Sometimes, morning meetings end before PMV reaches the control target (-0.5). The reason is that the whole building is thermally uncomfortable early in the morning because heat pump is off or setpoint temperature is 12°C at night. When outdoor air temperature is low, meeting may end before the whole building is warmed up. Thus, occupants can feel cold and uncomfortable during the whole meeting.

As indicated in table 5.1-4, for both alternatives, the condition in the classroom is similar to that in the meeting room. However, when we lower setpoint temperature to 12°C, the percentage of PMV within comfort range rises from 45% to 47% in the classroom, which seems strange. After we carefully check the PMV change for the whole month, we figure out that the time period of PMV exceeding 0.5 decreases, which increases the percentage of PMV within comfort range, as indicated in table 5.1-5. As what we control is heat pump, while heat pump is switched off during that period, thus our control logic cannot impact PMV during that period. As PMV is very close to the upper bound of comfort range and fluctuations of PMV can occur during several simulations, it’s highly possible that the decrease of the percentage of PMV within comfort range is incurred by the uncertainty of simulations.

Table 5.1-2 Comparison between baseline and two alternatives for open office

Baseline (Setpoint as 18°C at night)		Blue line: For warmer zones, PMV reaches control target (-0.5) at between 8.00 and 8.30 except for extreme conditions.
		Red line: For chiller zones, PMV reaches control target (-0.5) at between 9.30 and 10.20 except for extreme conditions.
		PMV takes least time to reach target. Then it fluctuates around certain value ranging from -0.7 to -0.5.
Heat pump off at night		Blue line: For warmer zones, PMV reaches control target (-0.5) at between 9.30 and 10.20 except for extreme conditions.
		Red line: For chiller zones, PMV reaches control target (-0.5) at between 10.40 and 14.00 except for extreme conditions.
		PMV takes most time to reach target. Then PMV pattern is similar to baseline.

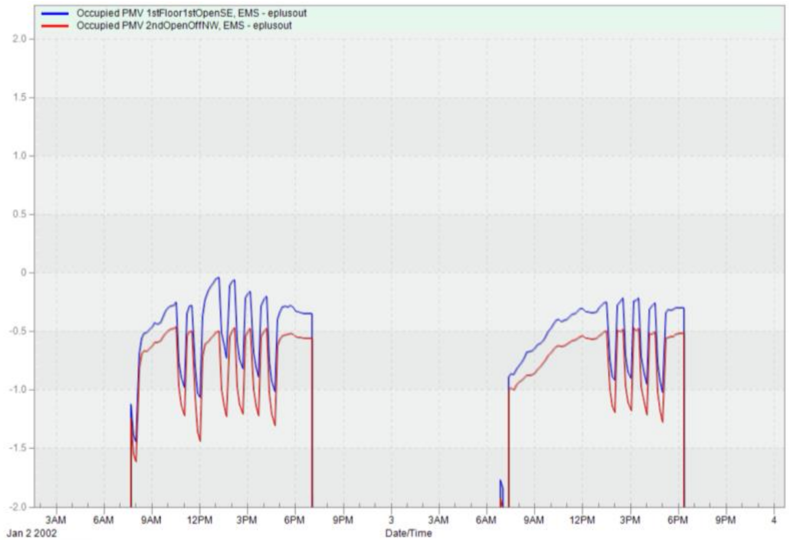
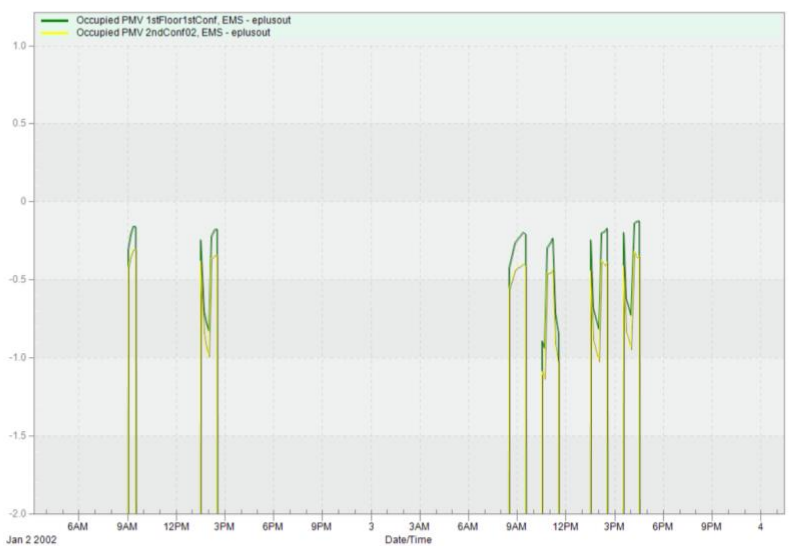
<table>
<tr><td rowspan="3">Setpoint as
12°C at night</td><td rowspan="3">
</td><td>Blueline: For warmer zones, PMV reaches control target (-0.5) at between 8.30 and 9.50 except for extreme conditions.</td></tr>
<tr><td>Redline: For chiller zones, PMV reaches control target (-0.5) at between 10.10 and 13.30 except for extreme conditions.</td></tr>
<tr><td>Time for PMV to reach target lies in the middle of three. Then PMV pattern is similar to baseline.</td></tr>
</table>

Table 5.1-3 Comparison between baseline and two alternatives for conference room

<table>
<tr><td rowspan="2">Baseline
(Setpoint as
18°C at night)</td><td rowspan="2">
</td><td>PMV takes least time to reach the target (-0.5).</td></tr>
<tr><td>After reaching the target (-0.5), PMV fluctuates around certain value relating to outdoor air temperature, which is the highest of the three. Thus, the mean PMV is the highest of three.</td></tr>
</table>

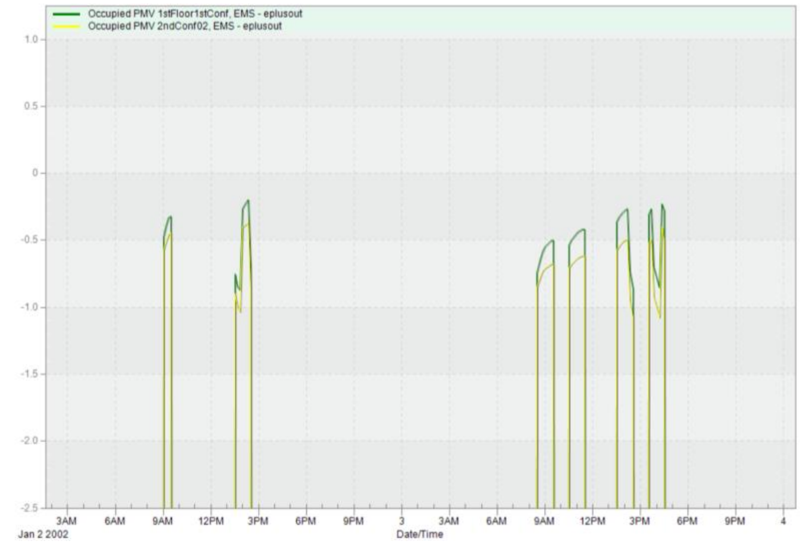
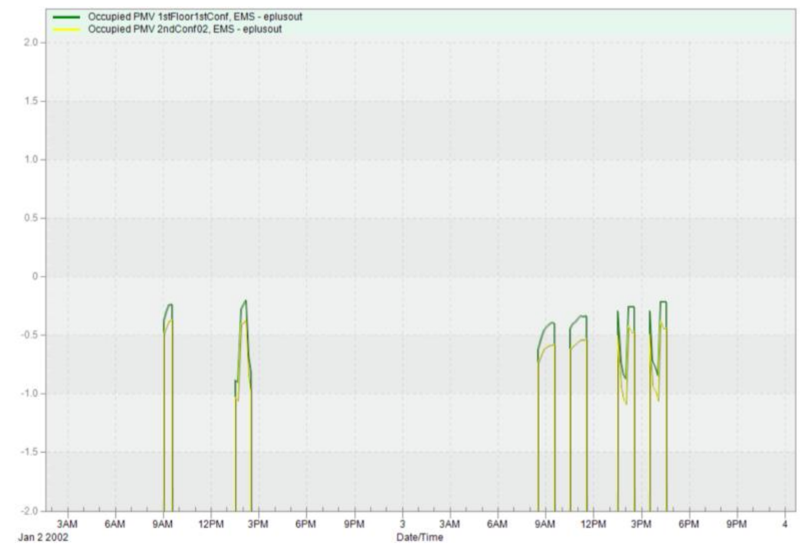
<table>
<tr><td rowspan="3">Heat pump off at night</td><td rowspan="3">
</td><td>PMV takes most time to reach the target (-0.5).</td></tr>
<tr><td>After reaching the target (-0.5), PMV fluctuates around certain value, which is the loweest of the three. Thus, the mean PMV is the lowest of three.</td></tr>
<tr><td>Sometimes, meetings end before PMV reaches target (-0.5).</td></tr>
<tr><td rowspan="3">Setpoint as 12°C at night</td><td rowspan="3">
</td><td>Time for PMV to reach target lies in the middle of three.</td></tr>
<tr><td>After reaching the target (-0.5), PMV fluctuates around certain value, which lies in the middle of the three. Thus, the mean PMV lies in the middle of three.</td></tr>
<tr><td>Sometimes, meetings end before PMV reaches target (-0.5).</td></tr>
</table>

Table 5.1-4 Comparison between baseline and two alternatives for classroom

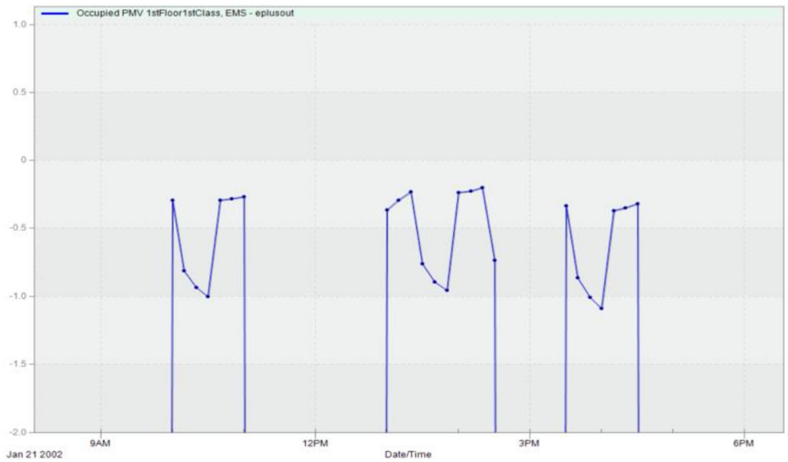
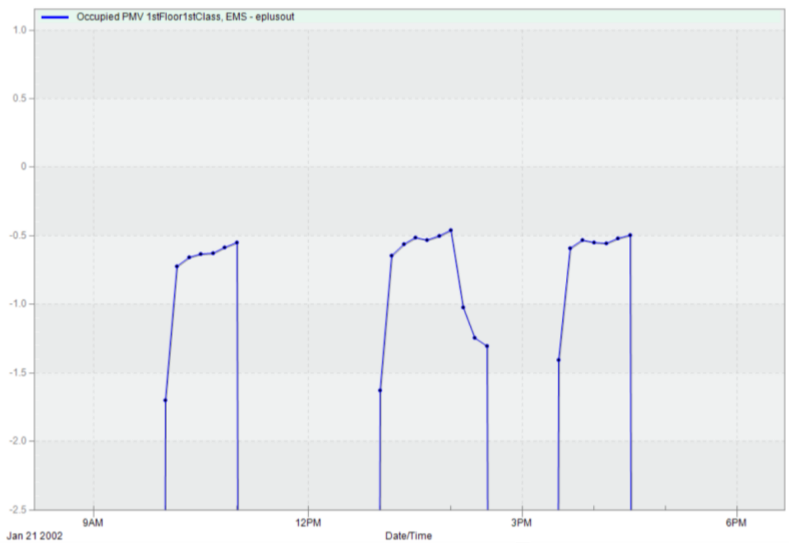
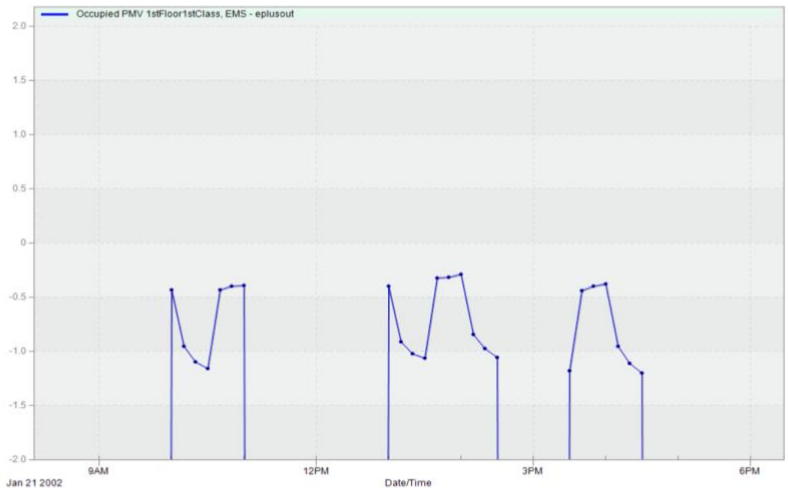
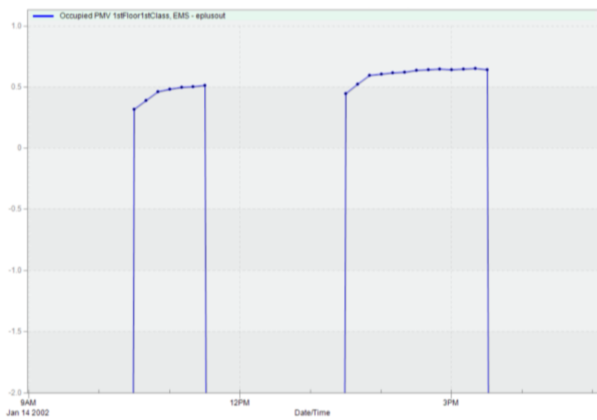
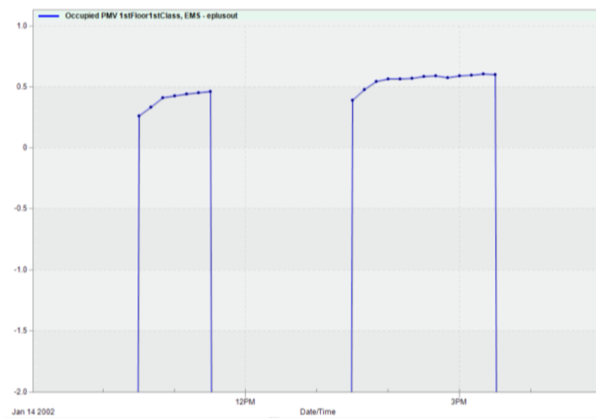
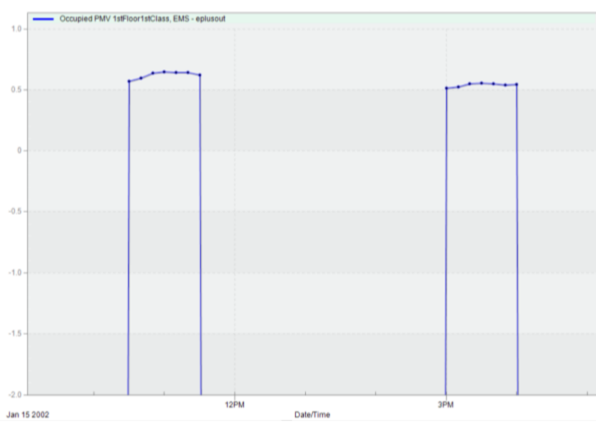
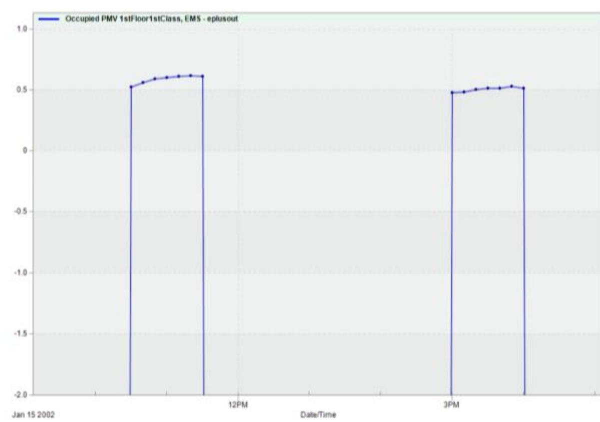
Baseline (Setpoint as 18°C at night)		PMV takes least time to reach the target (-0.5). PMV fluctuates around -0.5. The mean PMV is the highest of three.
Heat pump off at night		PMV can hardly reach the target (-0.5). The mean PMV is the lowest of three.
Setpoint as 12°C at night		PMV takes more time to reach the target (-0.5) than baseline does. PMV fluctuates around -0.7. The mean PMV lies in the middle of three.

Table 5.1-5 Comparison between baseline and lower setpoint scenario when outdoor air temperature is high

Baseline	Lower Setpoint to 12°C
	
	

5.1.3. Final Scenario

After carefully analyzing the baseline and two alternatives, we can know that in this case, we can do nothing during the period when PMV exceeds 0.5, however, we can make some changes when PMV drops below -0.5. For baseline and two alternatives, for most of time when occupants experience thermal discomfort, PMV fluctuates around -0.5 (baseline) or certain value below -0.5 (two alternatives). This means that most discomfort incurred by the relatively low indoor air temperature, which has something to do with the control logic. The original logic is that when indoor air temperature is under 18°C or minimum PMV is under -0.5, switch on heat pump. The temperature threshold is fine so we change the PMV threshold. The refined logic is that at night, lower temperature setpoint to 12°C and for the rest of day, when indoor air temperature is under 18°C or mean PMV is under 0, switch on heat pump. In this scenario, for open office and conference room, the percentage of PMV within comfort range rises from 67.4% to 75.4% and for classroom, the percentage of PMV within comfort range rises from 45% to 64%. The percentage of PMV within comfort range in each zone is shown in figure 5.1-14. Detailed information about PMV in each zone is summarized in Table 5.1-6.

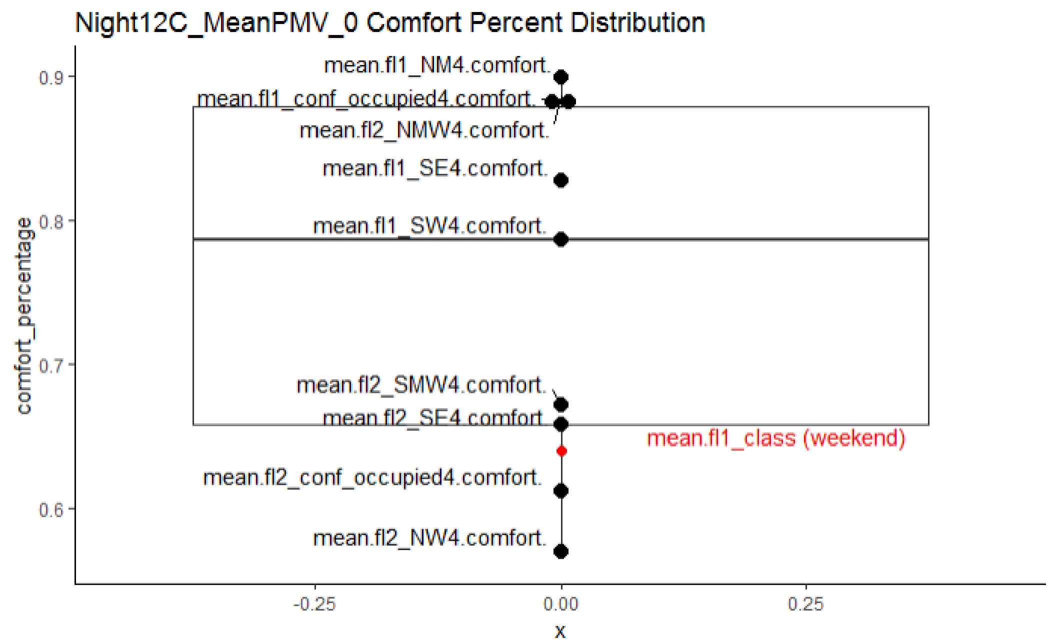


Figure 5.1-14 Final Scenario Comfort Percent Distribution

Table 5.1-6 Summaries of information about PMV in each zone (final scenario)

PMV	**1_conf**	**1_class**	**1_NM**	**1_SW**	**1_SE**	**2_conf**	**2_SMW**	**2_SE**	**2_NMW**	**2_NW**
Min	-0.845	-0.896	-1.7891	-2.1098	-2.0525	-0.999	-2.2726	-2.2854	-1.8131	**-2.2273**
1st Q	-0.355	-0.288	-0.2011	-0.4209	-0.3551	-0.594	-0.5430	-0.5551	-0.2500	**-0.6537**
Median	-0.202	-0.106	-0.0282	-0.2608	-0.1905	-0.431	-0.3764	-0.3878	-0.0684	**-0.4613**
Mean	-0.173	0.041	-0.0350	-0.2279	-0.1724	-0.389	-0.3392	-0.3486	-0.0790	**-0.4503**
3rd Q	-0.021	0.545	0.1608	-0.0024	0.0414	-0.240	-0.1131	-0.1204	0.1029	**-0.2517**
Max	**0.788**	**0.681**	**0.8525**	**0.8364**	**0.8547**	**0.615**	**0.7934**	**0.7934**	**0.8048**	**0.6198**

When PMV rises above 0.5, the maximum PMV (0.8547) occurs at the southeastern part of the open office on the first floor and it exceeds the upper bound (0.5) by 0.3547, which means occupants will experience low level of discomfort but not much.

Major time period when PMV for open office exceeds 0.5 falls between Jan 13th and Jan 15th. Only for a few certain time periods on Jan 12th, Jan 17th and Jan 18th, PMV for open office area on the first floor exceeds 0.5 by 0.13 at most. Besides, these periods last for 30 minutes at most. Thus, we focus on the analysis of major time periods, especially for Jan 13th.

As shown in figure 5.1-15, for the open office area, except the northwestern area on the second floor, the discomfort time period generally starts between 10.50 and 11.10 on Jan 13th, and it ends at 18.50 on Jan 13th. Occupants there will feel slightly uncomfortable for about 8 hours, almost one work day. For the northwestern area on the second floor, the discomfort time period starts at 13.50 and ends at 17.50, lasting for 4 hours.

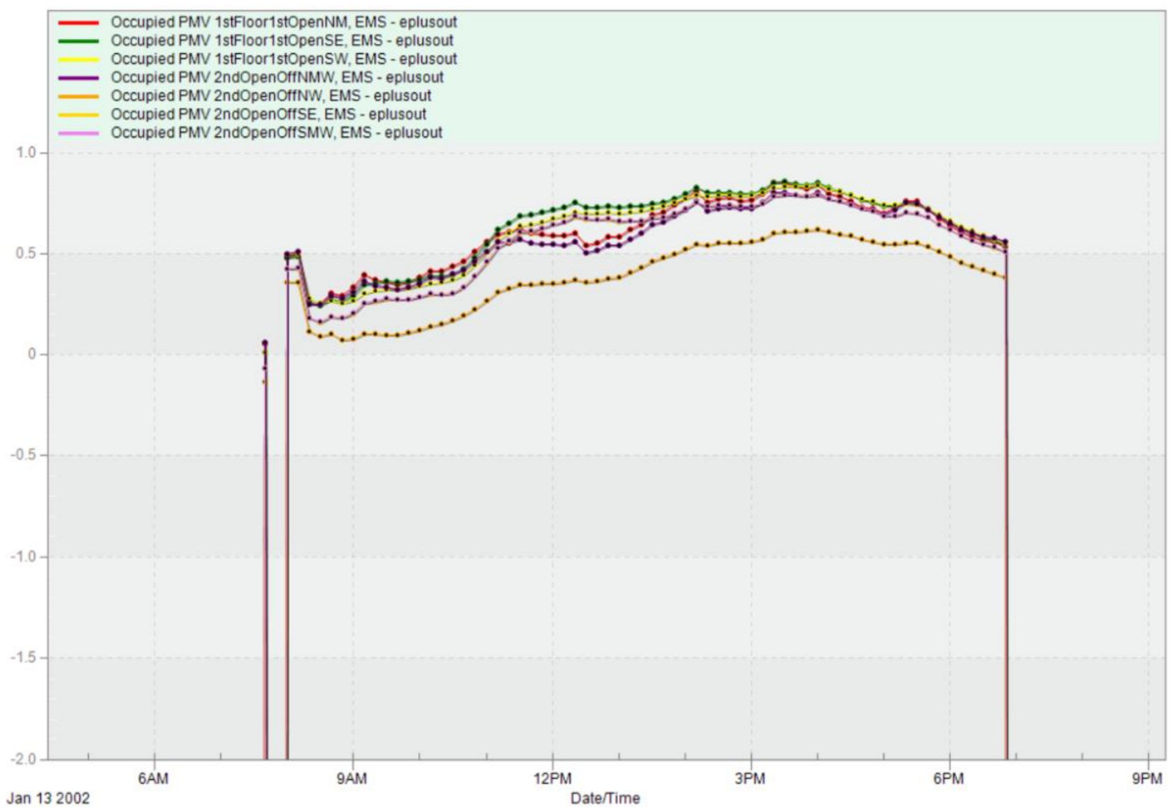


Figure 5.1-15 PMV fluctuation for open office on Jan 13th (final scenario)

As shown in figure 5.1-16, for conference room on the first floor, PMV exceeds 0.5 during the following time period on Jan 13th:

- From 14.00 to 15.00
- From 16.00 to 17.00

For conference room on the second floor, PMV exceed 0.5 during the following time period on Jan 13th:

- From 14.20 to 15.00
- From 16.00 to 17.00

Overall, for both conference rooms, so long as occupants there start to feel uncomfortable, the period can last for 1 hour.

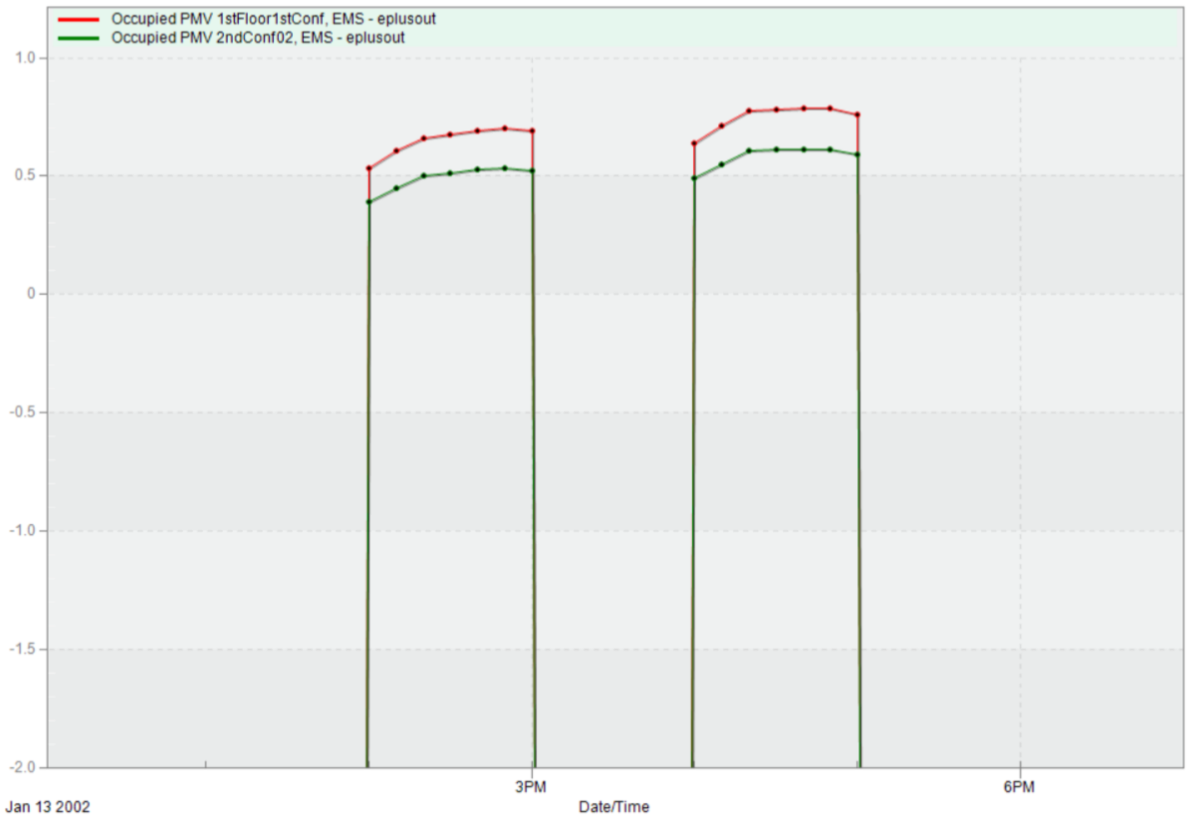


Figure 5.1-16 PMV fluctuation for conference room on Jan 13th (final scenario)

As shown in figure 5.1-17, PMV for classroom exceeds 0.5 during the following period:

- From 10.50 to 11.30 on Jan 14th
- From slightly later than 13.30 to 15.30 on Jan 14th, almost the whole class
- From 10.30 to 11.30 on Jan 15th, the whole class
- From15.00 to 16.00 on Jan 15th, the whole class

Generally, when occupants there will feel uncomfortable during the whole class, up to about 2 hours, except for the first class on Jan 14th.

As mentioned before, in this case, heat pump only provides heating, thus when PMV exceeds 0.5, the most we can do is to switch off heat pump. And during the periods when PMV exceeds 0.5, heat pump is already switched off, thus changing control logic will not impact PMV during these periods. As a result, the condition in this scenario is similar to that in baseline scenario. Tiny differences are incurred by the uncertainty of simulations.

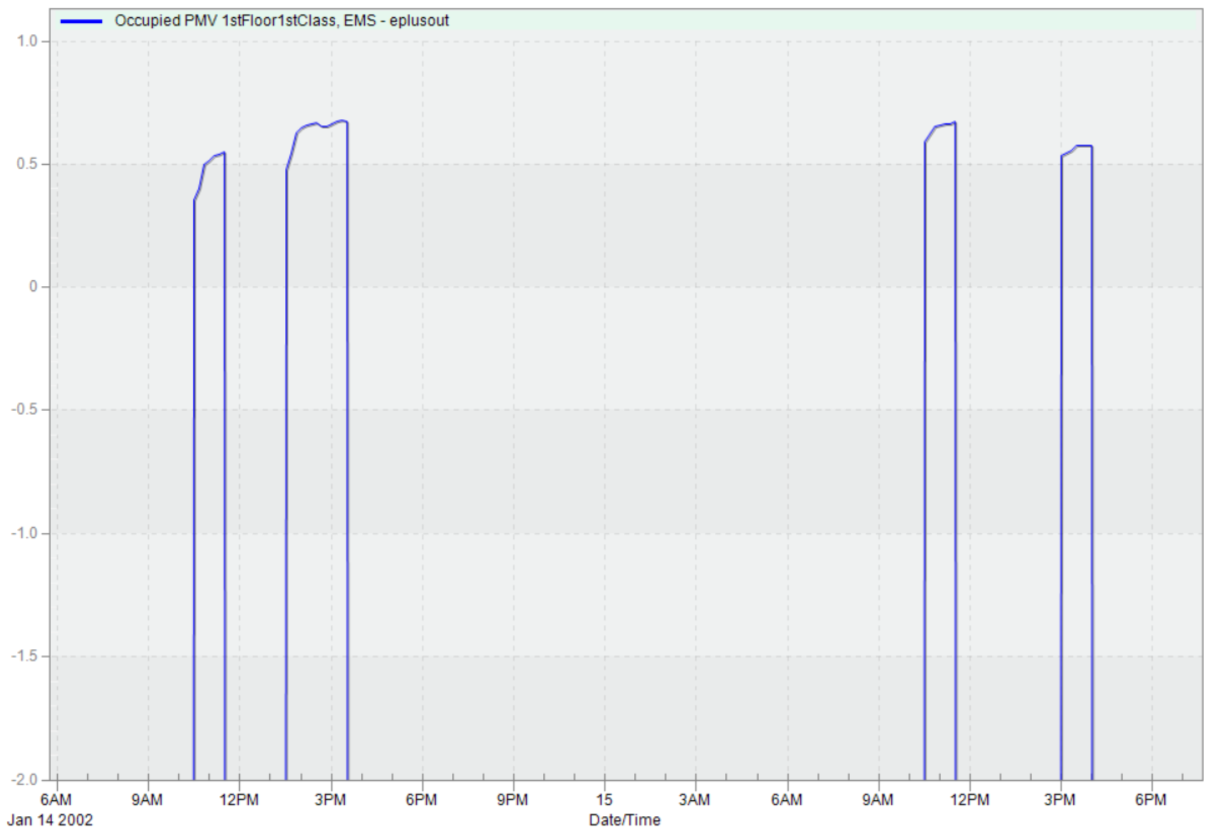


Figure 5.1-17 PMV fluctuation for classroom on Jan 13th (final scenario)

When PMV drops below -0.5, the minimum PMV (-2.2854) occurs at the southeastern part of the open office on the second floor and it is -1.7854 less than the lower bound (-0.5), which means occupants can experience high level of discomfort due to the low temperature.

As shown in figure 5.1-18, for final scenario, there are two typical patterns of PMV in open office.

- When outdoor air temperature is above -5°C (left pattern), mean PMV will rise to control target (0), and then fluctuate around 0. The reason is that when mean PMV reaches the control target (0), heat pump is switched off and on alternatively every 30 minutes. As the parameter is mean PMV, PMV for each zone is different, but it has similar pattern. PMV for the northwestern part of open office on the second floor rise to comfort range and then fluctuate around -0.5. PMV for other open office zones generally stays within the comfort range except early in the morning. The heating time last for 3 hours at most.

- When outdoor air temperature is under -5°C and above -10°C (right pattern), mean PMV can hardly reach the control target (0), so the heat pump is always on, and PMV for open office is always in comfort range except early in the morning. The heating time last for 3 hours at most.

As shown in figure 5.1-19, when outdoor air temperature is under -10°C on Jan 4th and Jan 5th (even under -15°C), mean PMV can never reach the control target (0), so the heat pump is always on. However, due to the extreme coldness outside, PMV for the northwestern part of open office on the second floor never reaches -0.5, which means occupants will feel cold and uncomfortable all day. PMV for other zones can be under -0.5 in the morning and late in the afternoon. Total time period can be about 6 hours at most.

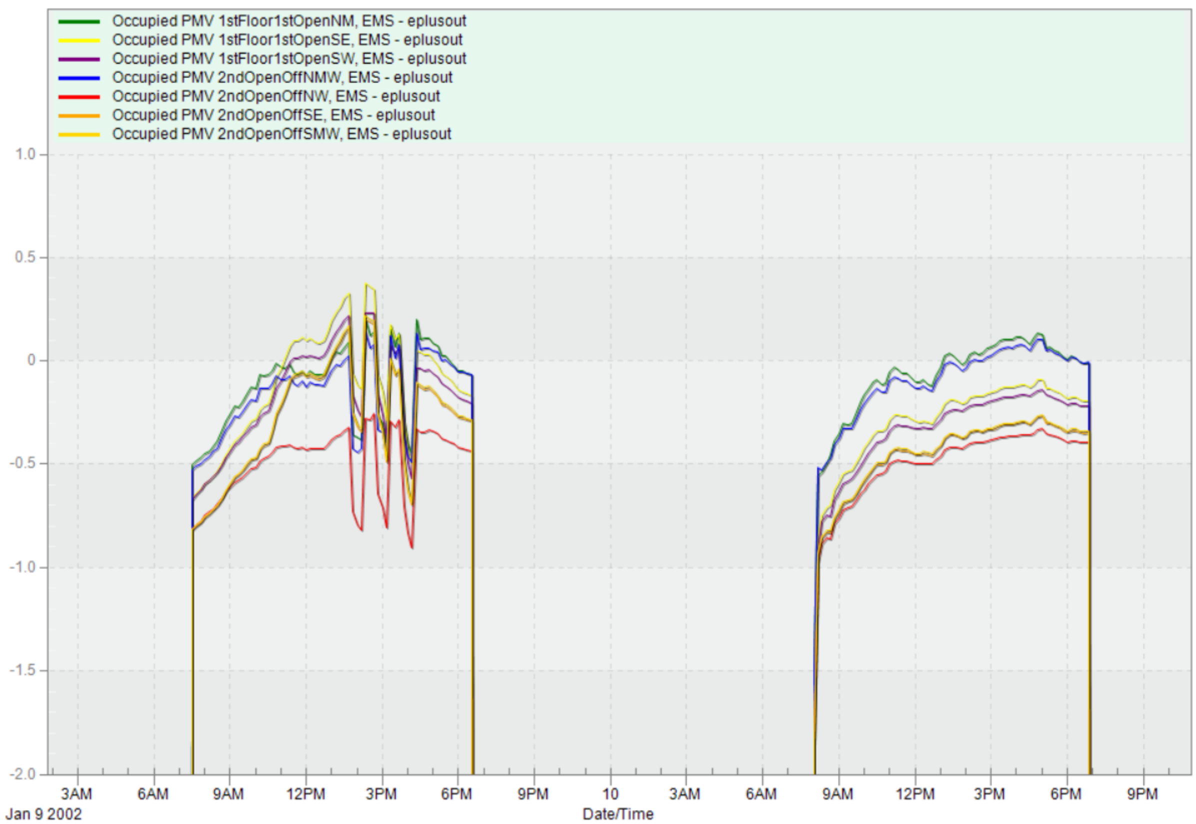


Figure 5.1-18 Typical patterns of PMV when occupants feel uncomfortable in open office (final scenario)

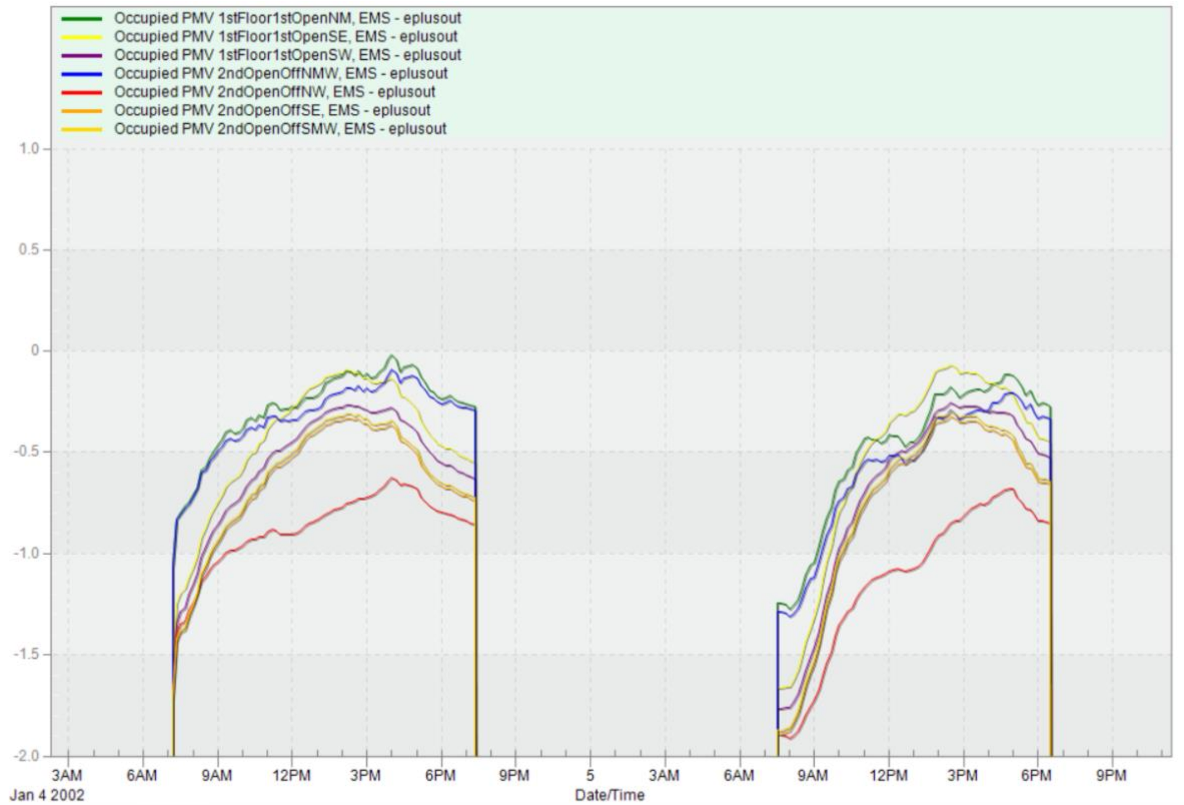


Figure 5.1-19 Pattern of PMV for open office during extreme cold periods (final scenario)

As shown in figure 5.1-20, patterns of PMV for conference room is similar to that for open office.

- When outdoor air temperature is under 0°C (left pattern), mean PMV cannot reach the control target (0) during the meeting, and heat pump is always on and PMV may reach -0.5 and enter the comfort range. When outdoor air temperature is extremely low, PMV cannot reach -0.5 during the meeting, which makes occupants feel cold and uncomfortable during the whole meeting.

- When outdoor air temperature is above 0°C, mean PMV will rise to control target (0), and then fluctuate around 0. PMV for conference room is above -0.5 during the majority of occupied period.

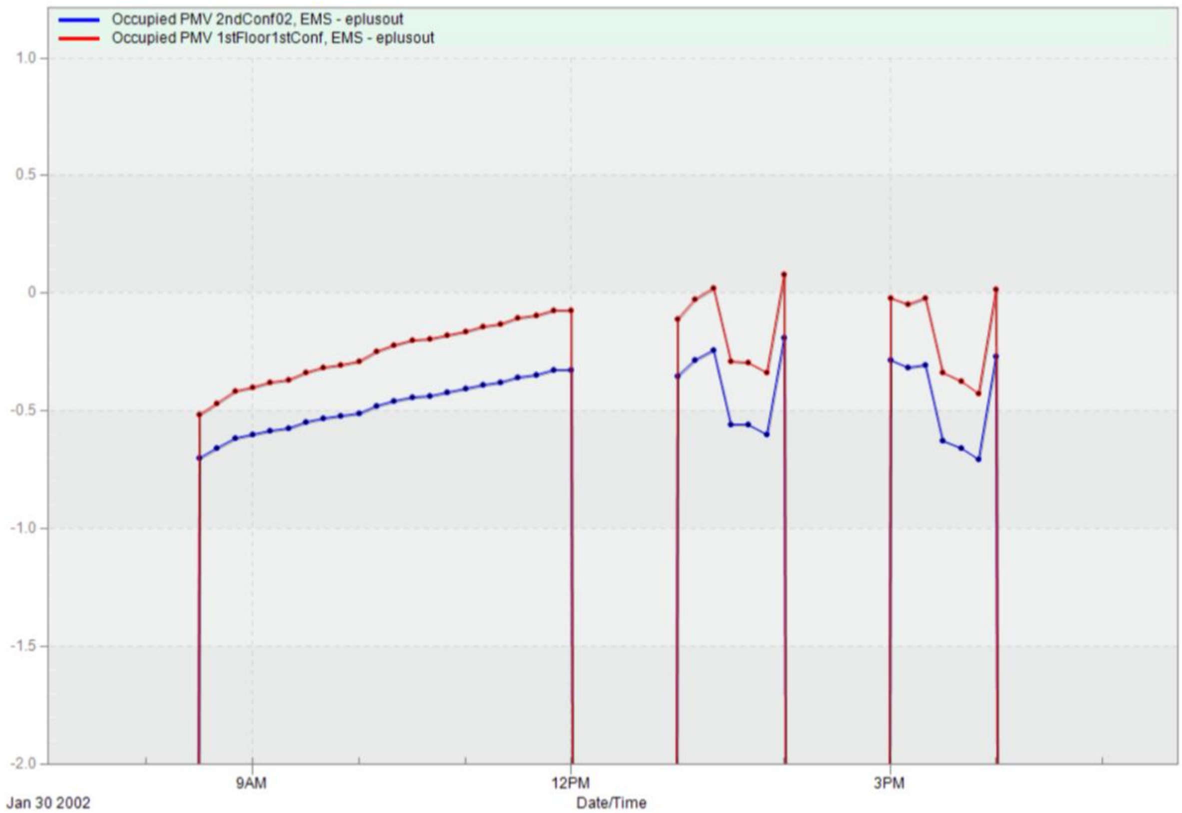


Figure 5.1-20 Typical patterns of PMV when occupants feel uncomfortable in conference room (final scenario)

As shown in figure 5.1-21, PMV in classroom is under -0.5 only during the following period on Jan 7th:

- From 10.30 to 10.50
- From 14.50 to 15.10
- From 14.00 to 14.05

For most of the class time, occupants feel comfortable. If occupants experience discomfort, discomfort period last for 20 minutes at most. However, the overall percentage of PMV within comfort range is 64%. Discomrt is largely incurred by the relatively high outdoor air temperature on Jan14th and Jan 15th.

Figure 5.1-21 PMV pattern in classroom when occupants feel uncomfortable (final scenario)

5.2. HVAC energy performance – Total heating energy consumption

Aside from PMV, energy consumption is the other index that we should consider, as good adjustments of control logic should improve overall thermal comfort without increasing energy consumption. The major purpose of switching off heat pump or lowering setpoint temperature to 12°C at night is to reduce heating energy consumption.

As shown in table 5.2-1, when switching off heat pump at night, heating energy consumption drops 35.7% compared to baseline, from 13926 kWh to 8959 kWh. When lowering setpoint temperature to 12°C at night, heating energy consumption drops 16.7% compared to baseline, from 13926 kWh to 11607 kWh. For these two scenarios, heating energy consumption does drop, especially when switching off heat pump, however, overall thermal comfort is compromised.

Table 5.2-1 Energy consumption and thermal comfort for four scenarios

Scenario	Energy consumption (kWh)	Reduction of energy consumption	% of thermal comfort in open office & conference room	% of thermal comfort in classroom
Baseline	13926	-	67.4%	45%
Heat pump off	8959	35.7%	46.7%	36%
Setpoint as 12°C	11607	16.7%	60.5%	47%
Setpoint as 12°C & Mean PMV as 0	12366	11.2%	75.4%	64%

Therefore, based on the 'lower setpoint temperature to 12°C at night' scenario, we change the control logic of switching on heat pump into that switch on heat pump when mean PMV is under 0 or indoor air temperature is under 18°C. And in this scenario, heating energy consumption drops 11.2% compared to baseline, from 13926 kWh to 12366 kWh. Meanwhile, overall thermal comfort improved.

5.3. Limitations

When we lower setpoint temperature to 12°C at night, and switch on heat pump when mean PMV is under 0 or indoor air temperature is under 18°C, we improve percentage of time that is thermally comfortable in open office and conference room by 8% (from 67.4% to 75.4%) and that in classroom by 19% (from 45% to 64%). However, there are certain limitations:

- In this case, heat pump only provides heating, thus we can do nothing when PMV exceeds 0.5 (approximately when outdoor temperature is above 15°C).
- In order to reduce energy consumption, we lower setpoint temperature to 12°C at night. As heating the building dose take time, thermal comfort is compromised early in the morning, even we start to heat the building at 5.00. When temperature is above -10°C, it takes up to 5.5 hours (from 5.00 to 10.30) to warm the whole building up so that the whole building is thermally comfortable.
- Even we change the control logic and make the building thermally comfortable for most of the work time, excluding early morning, there still exists short periods of time when PMV of some zones drops below -0.5. This is incurred by the short cycling of heat pump. When mean PMV reaches 0 or indoor air temperature reaches 18°C, heat pump is switched off and then it takes at least 30 minutes to switch on heat pump again. In 30 minutes after

heat pump is switched off, depend on the outdoor air temperature, PMV of some zones can drop below -0.5, which means these zones are slightly thermally uncomfortable.

- Due to the limited heating capacity, when outdoor air temperature is extremely low, PMV of some zones cannot reach -0.5 (on Jan 4th and 5th, northwestern part of open office on the second floor is always thermally uncomfortable even when heat pump is continuously on) or maintained above -0.5 (on Jan 4th and 5th, several zones are thermally uncomfortable late in the afternoon when outdoor air temperature drops).
- Due to the simplicity of this control logic, we cannot make individual control over each zone, which makes percentage of time when single zone is thermally comfortable varies from 55% to 90%, as indicated in figure 5.1-14.

6 Conclusions

In the process of this study, the impact of the limits or thresholds applied to the state variables "Indoor Air Temperature (IAT)" and "Predicted Mean Vote (PMV)" were assessed in different stages. A dual control objective was set, to achieve a decrease in energy consumption as well as increase in energy consumption. Initially, systems were switched off during unoccupied periods. However, while this produced a decrease in energy, occupant comfort was compromised. This was probably due to the colder temperatures at the start of the day as the space would need more time to get heated up. In general, switching off systems at night during winter could cause a variety of problems, for which this approach was not chosen. Instead, a setback temperature schedule was applied, based on which energy savings were observed with a better comfort percentage.

In our proposed control logic, the scheduled set-back temperature was proposed in combination with a mean PMV constraint. By implementing this logic on the EnergyPlus simulation platform, it was observed that the energy consumption decreased by 11.2%, while the indoor thermal comfort measure went up by 8%. Hence, the selected control logic was successful in achieving the initial control objective.

7 Reference

[1] Fanger, P. O. (1970). *Thermal comfort: Analysis and applications in environmental engineering*. Teknisk Forlag.

8 Appendix

8.1 Appendix A – Rstudio analysis code

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------
----------- tidyverse 1.2.1 --

## v ggplot2 3.1.0       v purrr   0.3.2
## v tibble  2.1.1       v dplyr   0.8.0.1
## v tidyr   0.8.3       v stringr 1.4.0
## v readr   1.3.1       v forcats 0.4.0

## -- Conflicts ------------------------------------------------------------
----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(ggplot2)
library(ggrepel)

#import simulation results
baseline <- read.csv('Results/selected/eplusoutbase.csv')

#create new column - temperature difference between zone air temperature and
the setpoint
baseline <- baseline %>%
  mutate(delta_t_1_conf= zone_air_temp_1_conf - setpoint_1_conf,
         delta_t_1_class = zone_air_temp_1_class - setpoint_1_class,
         delta_t_1_NM = zone_air_temp_1_NM - setpoint_1_NM,
         delta_t_1_SW = zone_air_temp_1_SW - setpoint_1_SW,
         delta_t_1_SE = zone_air_temp_1_SE - setpoint_1_SE,
         delta_t_2_conf = zone_air_temp_2_conf - setpoint_2_conf,
         delta_t_2_SMW = zone_air_temp_2_SMW - setpoint_2_SMW,
         delta_t_2_SE = zone_air_temp_2_SE - setpoint_2_SE,
         delta_t_2_NMW = zone_air_temp_2_NMW - setpoint_2_NMW,
         delta_t_2_NW = zone_air_temp_2_NW - setpoint_2_NW)

#set unoccupied value as NA
baseline[baseline == -999] <- NA

#create new dataframe for the weekend occupied zone - classroom on 1st floor,
 and remove the unoccupied hours
wknd_occupied <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_1_class, zone_air_temp = zone_air_t
emp_1_class, setpoint = setpoint_1_class, delta_T = delta_t_1_class, HVAC_ele
c_demand, heating_coil_power, air_supply_temp) %>%
  na.omit()

#create new dataframes for each zone of the other 9 zones, and remove the uno
ccupied hours
fl1_conf_occupied <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_1_conf, zone_air_temp = zone_air_te
mp_1_conf, setpoint = setpoint_1_conf, delta_T = delta_t_1_conf, HVAC_elec_de
```

```
mand, heating_coil_power, air_supply_temp) %>%
  na.omit()

fl1_NM <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_1_NM, zone_air_temp = zone_air_temp
_1_NM, setpoint = setpoint_1_NM, delta_T = delta_t_1_NM, HVAC_elec_demand, he
ating_coil_power, air_supply_temp) %>%
  na.omit()

fl1_SW <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_1_SW, zone_air_temp = zone_air_temp
_1_SW, setpoint = setpoint_1_SW, delta_T = delta_t_1_SW, HVAC_elec_demand, he
ating_coil_power, air_supply_temp) %>%
  na.omit()

fl1_SE <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_1_SE, zone_air_temp = zone_air_temp
_1_SE, setpoint = setpoint_1_SE, delta_T = delta_t_1_SE, HVAC_elec_demand, he
ating_coil_power, air_supply_temp) %>%
  na.omit()

fl2_conf_occupied <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_2_conf, zone_air_temp = zone_air_te
mp_2_conf, setpoint = setpoint_2_conf, delta_T = delta_t_2_conf, HVAC_elec_de
mand, heating_coil_power, air_supply_temp) %>%
  na.omit()

fl2_SMW <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_2_SMW, zone_air_temp = zone_air_tem
p_2_SMW, setpoint = setpoint_2_SMW, delta_T = delta_t_2_SMW, HVAC_elec_demand
, heating_coil_power, air_supply_temp) %>%
  na.omit()

fl2_SE <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_2_SE, zone_air_temp = zone_air_temp
_2_SE, setpoint = setpoint_2_SE, delta_T = delta_t_2_SE, HVAC_elec_demand, he
ating_coil_power, air_supply_temp) %>%
  na.omit()

fl2_NMW <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_2_NMW, zone_air_temp = zone_air_tem
p_2_NMW, setpoint = setpoint_2_NMW, delta_T = delta_t_2_NMW, HVAC_elec_demand
, heating_coil_power, air_supply_temp) %>%
  na.omit()

fl2_NW <- baseline %>%
  select(time, out_DBT, out_RH, PMV = PMV_2_NW, zone_air_temp = zone_air_temp
_2_NW, setpoint = setpoint_2_NW, delta_T = delta_t_2_NW, HVAC_elec_demand, he
ating_coil_power, air_supply_temp) %>%
```

```
  na.omit()

#create new column indicating if the PMV is in the range of comfort
wknd_occupied <- wknd_occupied %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))


fl1_conf_occupied <- fl1_conf_occupied %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl1_NM <- fl1_NM %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl1_SE <- fl1_SE %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl1_SW <- fl1_SW %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl2_conf_occupied <- fl2_conf_occupied %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl2_NMW <- fl2_NMW %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl2_NW <- fl2_NW %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl2_SE <- fl2_SE %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

fl2_SMW <- fl2_SMW %>%
  mutate(comfort = if_else(PMV > 0.5 | PMV < -0.5, 0, 1))

#create new dataframe of comfort percent for all zones
wkd_occupied <- data.frame(mean(fl1_conf_occupied$comfort), mean(fl1_NM$comfo
rt), mean(fl1_SE$comfort),  mean(fl1_SW$comfort), mean(fl2_conf_occupied$comf
ort), mean(fl2_NMW$comfort), mean(fl2_NW$comfort), mean(fl2_SE$comfort), mean
(fl2_SMW$comfort)) %>%
  gather()

#average comfort percentage during weekday
wkd_comfort <- mean(wkd_occupied$value)
wkd_comfort

## [1] 0.6743329
```

```
#average comfort percentage during weekend
wknd_comfort <- mean(wknd_occupied$comfort)
wknd_comfort

## [1] 0.45

#total heating energy consumption
heating_energy <- mean(baseline$heating_coil_power) * 31 *24 / 1000
heating_energy

## [1] 13926.26

#plot distribution of the comfort percentage in different zones
wkd_occupied %>%
  ggplot(aes(x = 0, y = value)) +
  geom_boxplot() +
  geom_dotplot(binaxis = 'y', stackdir = 'center') +
  geom_text_repel(aes(x = 0, label = key, vjust = -0.4, hjust = 1.05))+
  geom_point(aes(x = 0, y = wknd_comfort), color = 'red', size = 2) +
  annotate(geom = 'text', x = 0.22, y = wknd_comfort + 0.01, label = 'mean.fl
1_class (weekend)', color = 'red') +
  ggtitle('Baseline Comfort Percent Distribution') +
  theme_classic()
```

Same code was used with different variables' names to analyze the results of other scenarios.

```
#dataset of total heating energy consumption for 4 scenarios
Heating_energy_consumption <- c(heating_energy, heating_energy2, heating_ener
gy3, heating_energy4)
Heating_energy_consumption

## [1] 13926.26  8958.69 11606.92 12365.91

#dataset of weekday comfort percentage for 4 scenarios
Wkd_comfort_percent <- c(wkd_comfort, wkd_comfort2, wkd_comfort3, wkd_comfort
4)
Wkd_comfort_percent

## [1] 0.6743329 0.4670576 0.6046910 0.7538564

#dataset of wekkend comfort percentage for 4 scenarios
wknd_comfort_percent <- c(wknd_comfort, wknd_comfort2, wknd_comfort3, wknd_co
mfort4)
wknd_comfort_percent

## [1] 0.45 0.36 0.47 0.64
```

```
#dataframe of all variables comparison of 4 scenarios
comparison <- data.frame(Heating_Consumption_kWh = round(Heating_energy_consu
mption, digit = 2), Wkd_comfort_percent, wknd_comfort_percent)
comparison

##   Heating_Consumption_kWh Wkd_comfort_percent wknd_comfort_percent
## 1                13926.26           0.6743329                 0.45
## 2                 8958.69           0.4670576                 0.36
## 3                11606.92           0.6046910                 0.47
## 4                12365.91           0.7538564                 0.64

#plot total heating energy consumption for 4 scenarios
comparison %>%
  ggplot(aes(x = c('1_Baseline', '2_Unoccupied_heatpump_off', '3_nightsetback
_12C','4_nightsetback_12C_meanpmv_0'), y = round(Heating_energy_consumption,
digit = 2))) +
  geom_point(size = 4, shape = 17) +
  geom_line(aes(group = 1)) +
  geom_text_repel(aes(label = round(Heating_energy_consumption, digit = 2), v
just = -0.3, hjust = -0.3)) +
  scale_x_discrete(labels = c('1_Baseline' = 'Baseline', '2_Unoccupied_heatpu
mp_off' = 'Unoccupied', '3_nightsetback_12C' = 'Night_12C', '4_nightsetback_1
2C_meanpmv_0' = 'MeanPMV0')) +
  theme_bw()
```

```
#new dataframe of comfort percentages for 4 scenarios
comfort_comparison <- data.frame(round(Wkd_comfort_percent, digits = 4),
round(wknd_comfort_percent, digits = 4)) %>%
  gather(category, comfort) %>%
  mutate(scenario = c('1_Baseline', '2_Unoccupied_heatpump_off',
'3_nightsetback_12C','4_nightsetback_12C_meanpmv_0', '1_Baseline',
'2_Unoccupied_heatpump_off',
'3_nightsetback_12C','4_nightsetback_12C_meanpmv_0'))

#plot comfort percentages for 4 scenarios
comfort_comparison$category <- factor(comfort_comparison$category, levels =
c('round.Wkd_comfort_percent..digits...4.',
'round.wknd_comfort_percent..digits...4.'), labels = c('Weekday', 'Weekend'))

comfort_comparison %>%
```

```
  ggplot(aes(y = comfort, x = scenario, color = category)) +
  geom_point(size = 4, shape = 18) +
  geom_line(aes(group = category)) +
  geom_text_repel(aes(x = scenario, label = comfort, vjust = -0.4, hjust =
1)) +
  scale_x_discrete(labels = c('1_Baseline' = 'Baseline',
'2_Unoccupied_heatpump_off' = 'Unoccupied', '3_nightsetback_12C' =
'Night_12C', '4_nightsetback_12C_meanpmv_0' = 'MeanPMV0')) +
  theme_bw()
```

```
#plot the summary table
knitr::kable(comparison)
```

8.2 Appendix B – Control logic Python code

```
ob is a python list with the following items:
 0: hour (0-23)
 1: min (0-59)
 2: weekday (0-7, 0 is Monday)
 3: outdoor air temperature (C)
 4: outdoor air RH (%)
 5: diffuse dolar radiation (W/m2)
 6: direct solar radiation (W/m2)
 7-16: IAT of 10 zones (C)
 17-26: IAT cooling setpoint of 10 zones (C)
 27-36: PMV of 10 zones (-999 if no occupied)
 37: DX heating coil electric demand (W)
 38: HVAC total electric demand (W)
```

8.2.1 Baseline Control Logic Python Code

```
import gym, eplus_env, logging

logger = logging.getLogger('Ctrl-Tester');
logger.setLevel(logging.DEBUG);
ch = logging.StreamHandler()
ch.setLevel(logging.DEBUG);
logger.addHandler(ch);
logger.info('Running the simulation test...')
logger.info('Environment warm-up may take time.')
env = gym.make('CSL-short-cycle-v1');
ob, is_terminal = env.reset();
logger.info('Environment warm-up is done.')
cycle_count = 9999;

while is_terminal == False:
# !!DO NOT change the above lines!!
# The following lines are the baseline control strategy
# You should implement your control strategy here
    all_pmvs = ob[27:37] # Get all PMVs
    eff_pmvs = []; # Remove -999 from the PMVs
    for pmv in all_pmvs:
        if pmv != -999:
            eff_pmvs.append(pmv);
    min_pmv = min(eff_pmvs) if len(eff_pmvs) > 0 else 0; # The min occupied PMV
    all_iats = ob[7:17];
    min_iat = min(all_iats); # The min IAT
    dx_status = 1 if ob[37] > 0 else 0; # DX heating on/off state
    act = dx_status;
    if cycle_count >= 3:
        # Turn on heating if the conditions allow
        if min_pmv < -0.5 or min_iat < 18:
            act = 1;
        else:
            act = 0;
    if act == dx_status:
        # Remember cycle number
        cycle_count += 1;
    else:
        cycle_count = 1;
    logger.info('This observation is: %s'%ob);
    ob, is_terminal = env.step([act])
# !!DO NOT change the following lines!!
env.end_env();
```

8.2.2 "Heat Pump Off during Unoccupied Hours" Control Logic Python Code

```
import gym, eplus_env, logging

logger = logging.getLogger('Ctrl-Tester');
logger.setLevel(logging.DEBUG);
ch = logging.StreamHandler()
ch.setLevel(logging.DEBUG);
logger.addHandler(ch);
logger.info('Running the simulation test...')
logger.info('Environment warm-up may take time.')
env = gym.make('CSL-short-cycle-v1');
ob, is_terminal = env.reset();
logger.info('Environment warm-up is done.')
cycle_count = 9999;

while is_terminal == False:
# !!DO NOT change the above lines!!
# The following lines are the baseline control strategy
# You should implement your control strategy here
    all_pmvs = ob[27:37] # Get all PMVs
    eff_pmvs = []; # Remove -999 from the PMVs
    for pmv in all_pmvs:
        if pmv != -999:
            eff_pmvs.append(pmv);
    min_pmv = min(eff_pmvs) if len(eff_pmvs) > 0 else 0; # The min occupied PMV
    all_iats = ob[7:17];
    min_iat = min(all_iats); # The min IAT
    dx_status = 1 if ob[37] > 0 else 0; # DX heating on/off state
    act = dx_status;
    if cycle_count >= 3:
        # Turn on heating if the conditions allow
        if min_pmv < -0.5 or (min_iat < 18 and len(eff_pmvs) != 0):
            act = 1;
        else:
            act = 0;
    if act == dx_status:
        # Remember cycle number
        cycle_count += 1;
    else:
        cycle_count = 1;
    logger.info('This observation is: %s'%ob);
    ob, is_terminal = env.step([act])
# !!DO NOT change the following lines!!
env.end_env();
```

8.2.3 "Night Time Setback as 12 oC" Control Logic Python Code

```
import gym, eplus_env, logging

logger = logging.getLogger('Ctrl-Tester');
logger.setLevel(logging.DEBUG);
ch = logging.StreamHandler()
ch.setLevel(logging.DEBUG);
logger.addHandler(ch);
logger.info('Running the simulation test...')
logger.info('Environment warm-up may take time.')
env = gym.make('CSL-short-cycle-v1');
ob, is_terminal = env.reset();
logger.info('Environment warm-up is done.')
cycle_count = 9999;

while is_terminal == False:
# !!DO NOT change the above lines!!
# The following lines are the baseline control strategy
# You should implement your control strategy here
    all_pmvs = ob[27:37] # Get all PMVs
    eff_pmvs = []; # Remove -999 from the PMVs
    for pmv in all_pmvs:
        if pmv != -999:
            eff_pmvs.append(pmv);
    min_pmv = min(eff_pmvs) if len(eff_pmvs) > 0 else 0; # The min occupied PMV
    all_iats = ob[7:17];
    min_iat = min(all_iats); # The min IAT
    dx_status = 1 if ob[37] > 0 else 0; # DX heating on/off state
    act = dx_status;

    time = ob[0];

    if cycle_count >= 3:
        #during the night
        if time > 19 or time < 5:
            #turn on heating
            if min_iat < 12:
                act = 1;
            else:
                act = 0;

        #during the day, turn on heating
        else:
            if min_pmv < -0.5 or min_iat < 18:
                act = 1;
            else:
                act = 0;


    if act == dx_status:
        # Remember cycle number
        cycle_count += 1;
    else:
        cycle_count = 1;
    logger.info('This observation is: %s'%ob);
    ob, is_terminal = env.step([act])
# !!DO NOT change the following lines!!
env.end_env();
```

8.2.4 Final Proposed Control Logic Python Code

```python
import gym, eplus_env, logging

logger = logging.getLogger('Ctrl-Tester');
logger.setLevel(logging.DEBUG);
ch = logging.StreamHandler()
ch.setLevel(logging.DEBUG);
logger.addHandler(ch);
logger.info('Running the simulation test...')
logger.info('Environment warm-up may take time.')
env = gym.make('CSL-short-cycle-v1');
ob, is_terminal = env.reset();
logger.info('Environment warm-up is done.')
cycle_count = 9999;

while is_terminal == False:
# !!DO NOT change the above lines!!
# The following lines are the baseline control strategy
# You should implement your control strategy here
    all_pmvs = ob[27:37] # Get all PMVs
    eff_pmvs = []; # Remove -999 from the PMVs
    for pmv in all_pmvs:
        if pmv != -999:
            eff_pmvs.append(pmv);
    mean_pmv = sum(eff_pmvs)/len(eff_pmvs) if len(eff_pmvs) > 0 else 0; # The min occupied PMV
    all_iats = ob[7:17];
    min_iat = min(all_iats); # The min IAT
    dx_status = 1 if ob[37] > 0 else 0; # DX heating on/off state
    act = dx_status;

    time = ob[0];

    if cycle_count >= 3:
        #during the night
        if time > 19 or time < 5:
            #turn on heating
            if min_iat < 12:
                act = 1;
            else:
                act = 0;

        #during the day, turn on heating
        else:
            if mean_pmv < 0 or min_iat < 18:
                act = 1;
            else:
                act = 0;


    if act == dx_status:
        # Remember cycle number
        cycle_count += 1;
    else:
        cycle_count = 1;
    logger.info('This observation is: %s'%ob);
    ob, is_terminal = env.step([act])
# !!DO NOT change the following lines!!
env.end_env();
```

8.3 Appendix C – Worklog

All team member discussed and identified the control problem as a group together.

Ruiji Sun:

- Python code and flow chart for control logic
- Final report - Control logic implementation and simulation

Suzy Li:

- R analysis for baseline and proposed control logic evaluation
- Final report - introduction, analysis and approach

Rashmi Baliga:

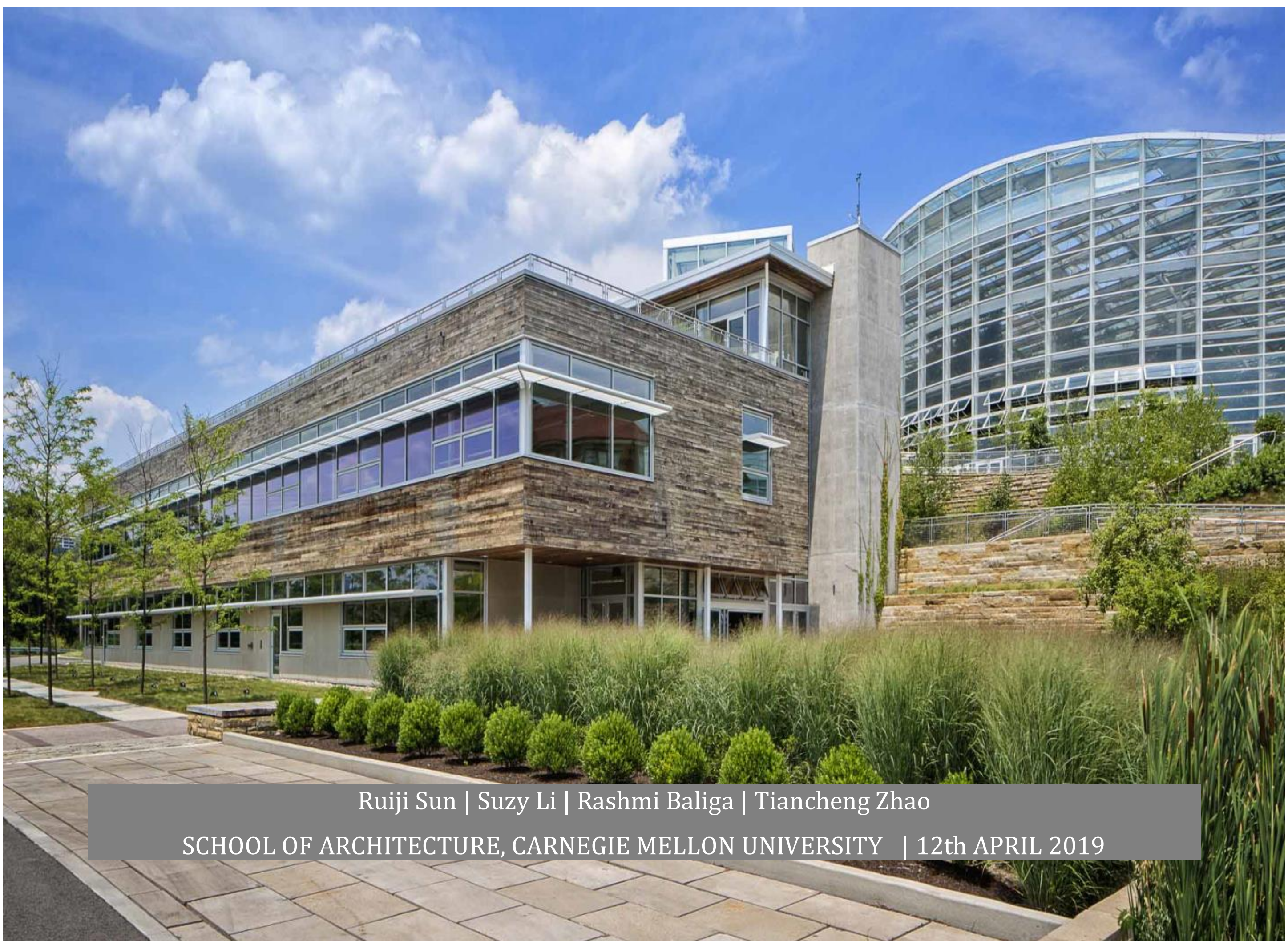
- Set up final report Template
- Final report - Executive summary and Control logic design

Tiancheng Zhao:

- Set up final report Template
- Final report - Comparative evaluation

Control System Design and Simulation for Building Systems - Part 2: Participatory Voting Control for thermal comfort optimization

Centre for Sustainable Landscapes Building



Ruiji Sun | Suzy Li | Rashmi Baliga | Tiancheng Zhao

SCHOOL OF ARCHITECTURE, CARNEGIE MELLON UNIVERSITY | 12th APRIL 2019

Executive Summary - Part2

This study is a continuation of previous studies of the Center for Sustainable Landscapes (CSL) building at Pittsburgh, Pennsylvania. Based on the Post Occupancy Evaluation conducted before, a few thermal comfort issues were identified. To improve the situation, a new control logic was needed. The aim of this study is to involve the building occupants as active participants in the temperature control system, rather than relying solely on calculated Predicted Mean Vote values (PMV).

Keeping this aim in mind, a literature review was conducted to identify relevant assumptions and previous studies which involved participatory sensing from occupant perception. The information obtained from this review was used to shape a schematic control logic diagram, which would then need to be implemented using Simulink. The implementation and working of the whole system relied heavily on the occupants' thermal comfort, which had to be obtained from each user.

To do so, a user interface was designed on Matlab App Designer, which could allow users to communicate their thermal sensation perception to the control system. The interface was designed to be intuitive, with minimal instructions and language based inputs. Another interface was designed to allow the facility managers to monitor the comfort status of the occupants, and override the system in case of system malfunctions or any other unreasonable situations. The data obtained was aggregated every ten minutes, and supplied to the control logic. The output of the aggregation logic was a setpoint temperature adjustment, which was then added to the existing setpoint temperature.

The logic was tested for a single zone, in the Northwest Region on the second floor of the CSL building. An EnergyPlus model was created, and the control logic was applied to that particular zone. Using co-simulation platform MLE+, the Simulink logic and the EnergyPlus model were coupled. Since only one zone was being analyzed, the outcome was assessed using the PMV values. The results of the simulation showed improved PMV values, closer to 0, as a result of using the proposed control logic. Since EnergyPlus provides energy consumption results for the entire building, and not for individual zones, the energy consumption could not be compared for the baseline and proposed logic. Limitations of the study and scope of work to be done in the future are also described in this report. Overall, the control logic seemed to work and provided the expected improvement in occupant comfort.

Table of Contents - Part2

1 Introduction

The aim of this study was to design and simulate a control system for a system of the Center for Sustainable Landscapes (CSL) building in Pittsburgh, PA. The building is located in a fairly isolated setting, next to the Phipps Conservatory. There is no surrounding density, or shading structures, but a lot of vegetation surrounds the structure.

For this study, the HVAC system was chosen based on observations during the Post Occupancy Evaluation findings in Assignment 2, and based on the characteristics of the existing building systems in Assignment 1 of this course. A literature review was then carried out to identify methods to improve occupant control over their thermal control in the office. Based on this analysis, a participatory voting control was proposed, and the details of this proposal are listed in this study.

1.1 Introduction to Building HVAC System

One of the largest energy sinks of the building is its Heating, Ventilation, and air-conditioning (HVAC) system. In commercial applications in US, HVAC systems account for 53.4% of the total energy use. [1] Within the CSL building, all spaces are mechanically heated nor cooled except the atrium. The mechanical system used in the building uses a geothermal source as a heating and cooling source for the HVAC system, which saves lots of energy. Most of the terminals are variable air volume (VAV) boxes located under the floor, and some of them are air diffusers located in the ceiling. The ventilation strategy for the CSL building incorporates natural ventilation strategies with this mechanical system. When temperature, humidity and air quality conditions are favorable for opening windows, mechanical systems are locked out to conserve energy.

2 Reflections on Post Occupancy Evaluation Analysis

2.1 Observations

A detailed Post Occupancy Evaluation (POE) was carried out during a previous study at the CSL building. Some of the key factors of discomfort for the occupants were found to be thermal and acoustic comfort, and glare issued from the glazed surfaces. Since this study not only proposed to address one of the identified problems, but also sought to propose a solution compatible with the existing system, the control strategy chosen involved the HVAC system. While acoustic disturbance and glare were problems, acoustic disturbance is not a control-addressable issue. Also, there are no existing motorized blinds or addressable louvers/ shading devices along the building's perimeter. Therefore, while glare was identified as a larger issue than thermal comfort, addressing thermal comfort was identified as a more practical and applicable solution.

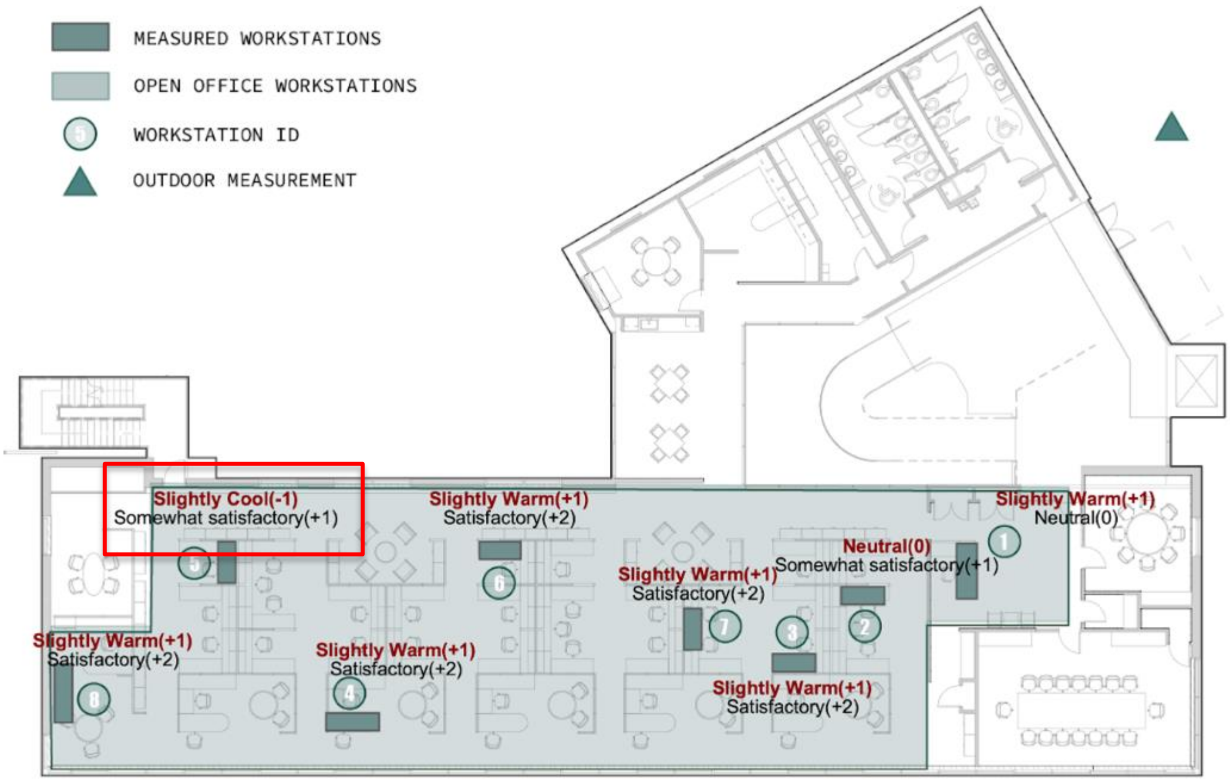


Figure 2.1.1 Thermal Sensation and Satisfaction Votes observed in the POE study

During the evaluation, the thermal comfort levels and satisfaction of the occupants was found to be varying, as can be seen in Figure 2.1.1. It can be seen that one of the occupants feels slightly cool even when most other people feel slightly warm. This shows a degree of variability between the perceptions of comfort by occupants, and between different thermal zones. For this reason, occupant control over the thermal conditions was studied in more detail.

Keeping this in mind, the existing characteristics of the HVAC control were noted. Of most interest was the current method of providing occupants control over their thermal conditions. This method is elaborated below.

2.2 Method and Limitations of Current Occupant Thermal Control System

Each zone has a thermostat mounted on a pole. This thermostat is easily accessible by occupants, but does not display the setpoint temperature. The thermostat has a lever built into its side, which can be pushed up or down to indicate whether the occupant wants the temperature to be warmer or cooler. The number of votes requesting the change is this evaluated. A picture of this thermostat is presented in Figure 2.2.1.

Figure 2.2.1 Thermostat with lever for occupant control, present in each thermal zone

2.2.1 Limitations

Some limitations of this implementation are listed below:

a. The thermostat is located in the open, with no authentication requirement or indication of the zone limits that it addresses. This means that any occupant, mistakenly or deliberately, has the ability to influence the voting pool of occupants, biasing their thermal preferences and affecting the satisfaction of the inhabitants of that zone.

b. Due to lack of authentication, it is also not possible to keep a tab on who is voting. The same user may vote for a certain temperature condition repeatedly, once again resulting in a bias.

c. The system does not provide any feedback, and is not linked in any way to support personal thermal comfort device recommendations.

Based on these limitations, and noting the importance of occupant control in buildings, an alternate thermal comfort system was proposed. This system is elaborated in section 2.3.

2.3 Proposed Occupant Thermal Control System

The proposed thermal control system aims at providing occupants the opportunity to provide individual feedback. Some characteristics of this system are:

a. There is a user interface for each user in a given zone. This interface has the functionality to enable the user to provide personalized feedback to the control logic. Individual votes are aggregated and communicated with the control system.

b. If a user votes multiple times, only the last vote in each duration is counted during the aggregation.

c. The interface can provide personal comfort device recommendations, for the user to either accept or deny.

d. All the collected votes are periodically aggregated, and a temperature change is computed. The computed temperature change is sent to the control system, where the zone setpoint temperature is modified to reflect occupant votes. This aggregation is not instantaneous, but is done every ten minutes to avoid continuous impact to the HVAC system.

The system is elaborated in greater detail in later sections. A conceptual sketch generated for this system is indicated in Figure 2.3.1.

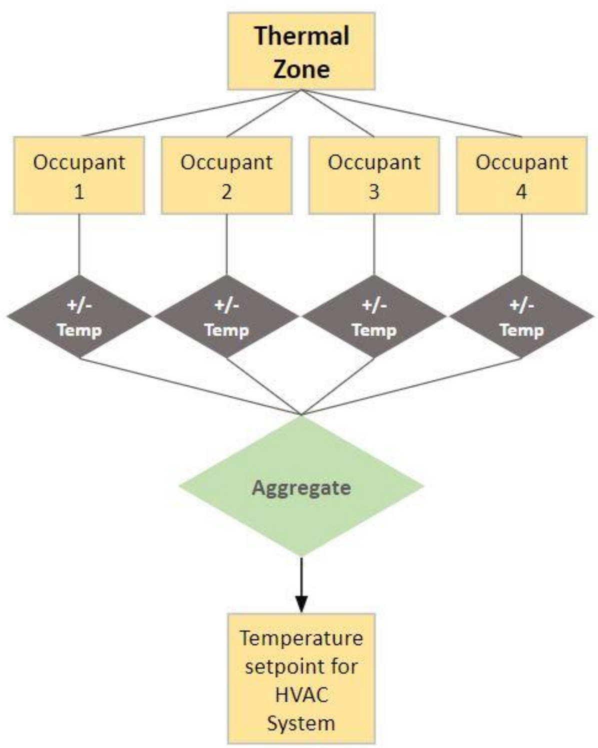


Figure 2.3.1 Conceptual Representation of the Proposed Control Strategy

2.3.1 Challenges

To implement this control logic, several challenges needed to be addressed. These are listed below:

a. Method of Aggregation: Multiple subjective votes needed to be aggregated following their collection. A method that is truly representative of majority of the votes in a given zone was required.

b. Method to Handle Heating and Cooling Requests in the Same Zone: Since simultaneous heating and cooling is not possible within the same thermal zone, a good way to satisfy majority of the people despite disparity in thermal sensations was needed.

c. Duration of Aggregation: How often would the aggregation occur?

d. Details of the User Interface: The degree of information included in the user interface for the occupants and the facility manager needed to be analyzed.

2.3.2 Literature Review

To create an informed control logic, a literature review of some relevant sources was conducted. The focus of the literature review was to identify methods to receive and interpret feedback from occupants in a building and provide it to a building control system. Some observations from these sources which shaped the proposal offered in this study are described in this section. When the review was started, there were four questions which needed to be answered –

(a) How often are the votes from users aggregated?

(b) How do individual votes get weighted during the aggregation?

(c) What could be the method to confirm the presence of the user/voter in the thermal zone where the vote is being cast?

(d) What can make up an effective interface?

A United States Patent application was first explored. Henneberger et al. discussed different methods to implement a temperature control voting system. [2] Each of their methods involved division of the building into zones. The systems are implemented for occupants in a particular zone. Each occupant is given the ability to communicate their vote to provide feedback to the HVAC control system. The process is divided into components such as a network, server, HVAC system and user devices. Different implementation alternatives were discussed.

In another study, Erickson and Cerpa implemented "Thermovote", a participatory thermal comfort voting strategy, over a period of five months. [3] In the study, four studies were conducted – a baseline evaluation, learned control schedule, real time control and long term real time control. In this process, iPhone and android applications as well as a website were developed to enable users to provide feedback. From the results of the real time study, which is related to our proposal, 100% satisfaction was observed, compared to a much lower satisfaction rate of 25% during the baseline evaluation. In addition, a 10.1% increase in energy savings was observed compared to the baseline.

From both these studies, the relevant observations to answer the aforementioned four questions are provided below. The application of these observations in the current proposal is also described below.

(a) How often are the votes from users aggregated?

In Erickson and Cerpa's implementation, the most recent set of votes in every 10 minute period was considered. In the current proposal, the same duration or time step for aggregation has been considered.

Since the patent application by Henneberger et al. was more general and conceptual, it did not discuss specific durations or time steps.

(b) How do individual votes get weighted during the aggregation?

All user votes were weighted equally in Erickson and Cerpa's study, while Henneberger et al. suggested that votes could vary based on the time of day as well as seniority. For the current proposal, votes were not segregated or differently weighted based on time of day or seniority. A different logic than these two was sought.

(c) What could be the method to confirm the presence of the user/voter in the thermal zone where the vote is being cast?

In Erickson and Cerpa's study, occupants had to submit a room request, which was manually verified during the study. The rooms were stored once the verification was complete. While this implementation worked due to the short duration of the study, in a long term implementation this could be tedious and inefficient. Henneberger et al., on the other hand, described using barcodes or QR codes to identify specific locations. They also talked about the use of GPS and WiFi to detect the location of occupants with respect to the pre-defined "bounds" of the thermal zone. In the final part of this report, for the web based implementation, QR codes were used. Initially, the control logic included an authentication step. However, due to limitations of time and knowledge, this step is not presented in the current implementation of our proposal.

(d) What can make up an effective interface?

Android applications, iOS applications, and a website were developed for the purpose of user interaction in Erickson and Cerpa's method. Henneberger et al. discussed the possibility of using a web interface, desktop application, or altogether replacing subjective opinions with bio-sensed data for more robust feedback free of psychological errors or problems. However, the accuracy of translating bio-sensed data to thermal comfort results has its own limitations. For this study, direct feedback from occupants was sought using a desktop based application, created using the matlab app designer application. In the final part of this report, a web based implementation was also set up.

In another study, Jazizadeh et al. used participatory and physical sensor data to generate a framework to learn occupant comfort profiles. [4] A mobile based application was used in this case as well, for the communication of thermal comfort preferences between the occupants and building systems. An algorithm was used to learn comfort profiles of the individual occupants, and the control logic aimed to modify the temperature in a zone as per the comfort profile of the occupants. In this study, the maximum temperature change in each step was limited to 1.5°F (0.83°C). The same maximum temperature change threshold was applied to the current study as well. For this reason, a multiplier of 0.5 is used while setting up the initial control logic. This is explained in more detail in the description of the control logic.

In another document containing a lecture from a Princeton University Course (cos 521: Advanced Algorithm Design), a weighted majority algorithm was studied to explore if weighting occupants whose votes resulted in a better PMV value could be possible. [5] However, the algorithm in this study was suitable for binary votes, where there was one correct vote and one incorrect vote for each voting iteration. Thermal comfort and PMV are very complex, and there is no single "correct" answer. Since the limits of the duration of this study as well as knowledge, a different algorithm was proposed, and the source was not utilized. However, it is recognized that a method of weighting more consistent voters higher could yield better results, and a similar source which is relevant could be explored in future studies.

2.3.3 Study Limitations Influencing the Design of the Control Logic and Interface

Although the control logic is proposed for the whole building, the virtual implementation and simulation have been designed for a single zone having five users, for the sake of simplicity and due to limitations of time and knowledge. The ideal implementation of this control logic would involve the different user interfaces linked through a network and connected to a central server. However, this was not possible in the current study. Instead, five different app screens were used for a live demonstration and implementation, using the local computer as a conceptual server.

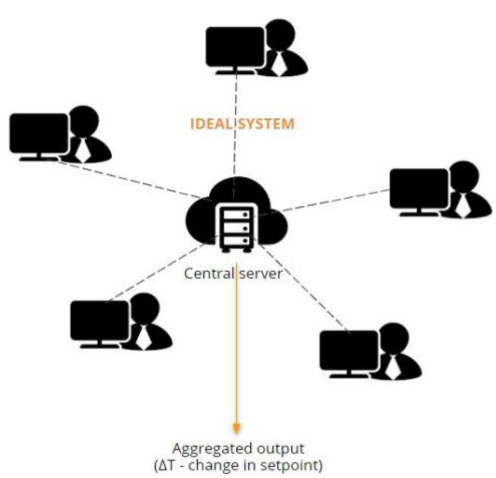


Figure 2.3.2 Conceptual representation of the ideal system to implement the proposed control logic

3 Control Logic Design

Based on the preliminary study and literature review discussed in previous sections, a control logic was designed to support the proposed system. The various details and the process arrived at are discussed in this section.

3.1 Control Objectives, State Variables and Control Variables

Control objective – To improve thermal comfort

State variables – The aggregation of user comfort votes and the time elapsed since the last update to the setpoint temperature are the two state variables.

Control Variables – The zone setpoint temperature acts as the control variable, which is updated based on the state variables.

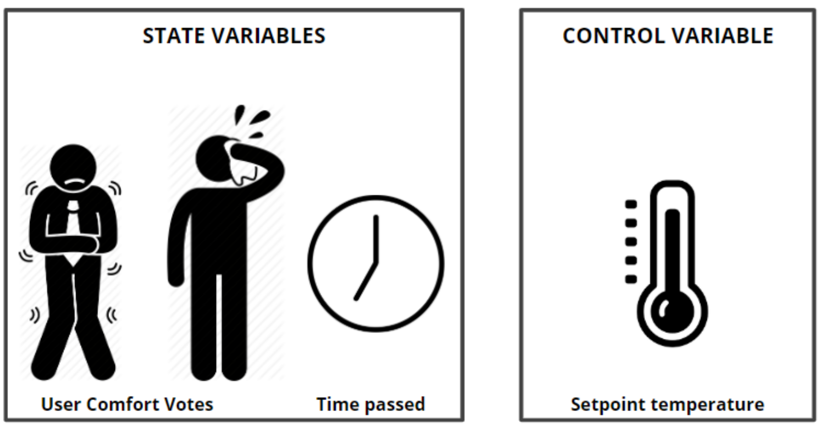


Figure 3.1.1State and control variables for the proposed control logic

Controllers: The simulink code containing the control logic acts as the controller in this study.

Actuators: Since the logic is tested using an EnergyPlus simulation, the EnergyPlus model acts as the actuator. If this were actually being implemented, the supply air temperature could be considered an actuator for the zone air temperature setpoint to be achieved.

Disturbances:

Though these are not quantified in the control logic, there could be several disturbances which could affect the proposed logic and its ability to achieve the desired results. Some of these are listed below:

1. Windows being left slightly open, causing draft and affecting the ability or duration in which the system can achieve the setpoint temperature.

2. Misrepresentation of thermal sensation by users in an effort to disproportionately weight their vote. For example, someone feeling mildly warm could vote "extremely hot" due to the incorrect perception that it will get quicker results.

3.2 Schematic Representation of Logic

3.2.1 General Flow of Logic

The general flow of logic followed in the proposed control method is provided in this section. The schematic diagrams in Figures 3.2.1, 3.2.2 and 3.3.3 represent the exact process followed, which was later modeled on Simulink.

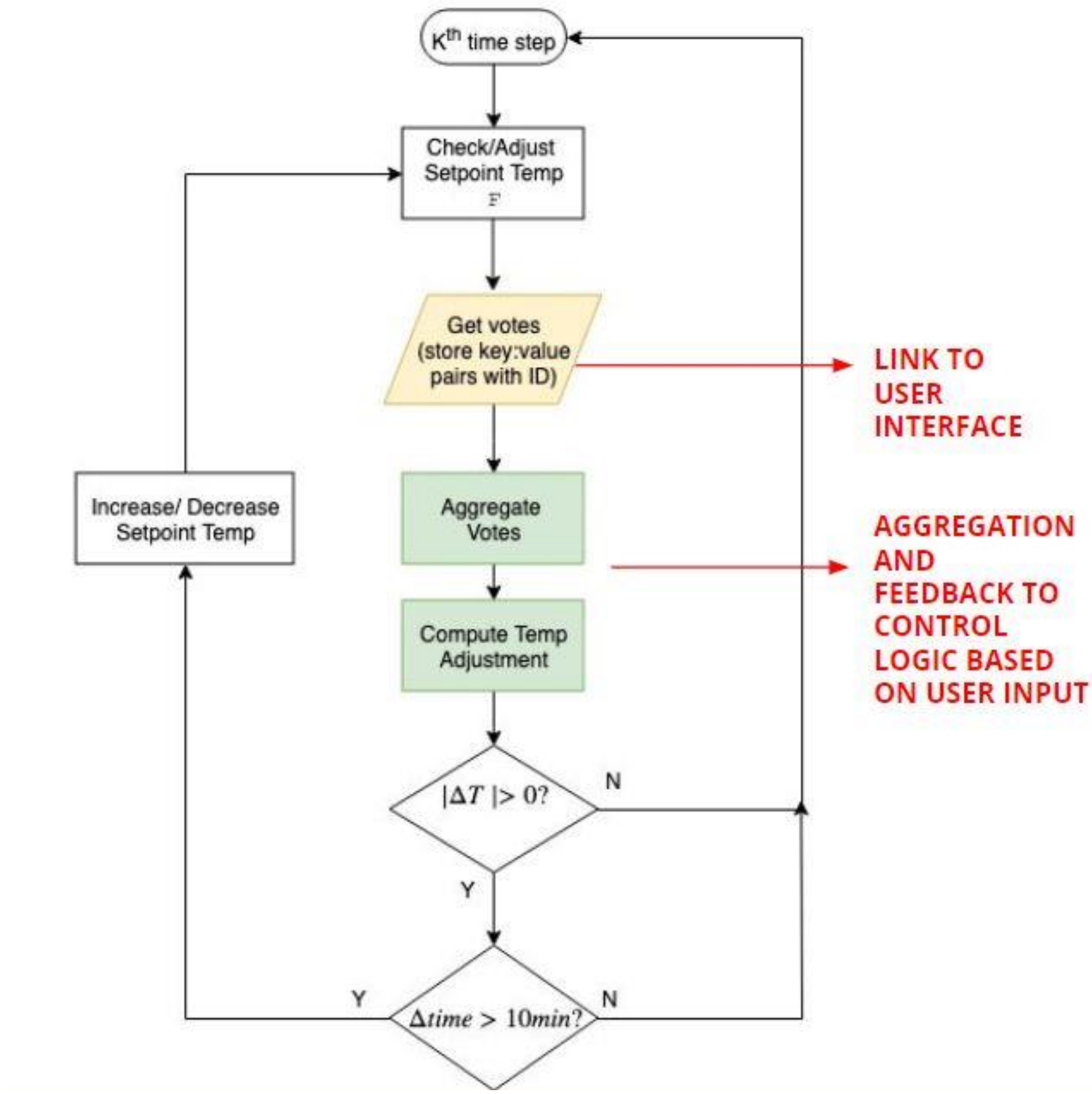


Figure 3.2.1General Flow of the Control Logic

The general flow of the control logic relies on a periodic collection of votes from the user. It is assumed that the votes obtained from the user are collected on a scale ranging from -3 to 3, representing the following sensations:

Table 3.2.1Relationship between thermal sensation/perception and numeric value on the scale

Number on the scale	Corresponding Thermal Sensation
-3	Very Cold
-2	Cold
-1	Cool
0	Neutral
+1	Warm
+2	Hot
+3	Very Hot

The actual representation of this scale for the end-users is elaborated later, in the User Interface Design section. These votes are collected from the user and aggregated. Based on the aggregated value, the required temperature adjustment is computed, and the new setpoint temperature is provided to the controller. The basic high level control logic can be written as follows

- *@some time step k:*
- *Check AND Update Setpoint Temperature IF required*
- *Get new votes from users*
- *Aggregate votes AND Compute Setpoint Adjustment*
- *IF $|\Delta T| > 0$ and time elapsed > 10 minutes:*
    - *Increase OR Decrease Setpoint Temperature*
- *Repeat @k+1*

The conceptual methods of collecting, storing and aggregating values is provided in the following paragraphs. The actual implementation of these methods in Simulink is described later in this report (Section 6). All Temperature values are represented in Fahrenheit, and converted to Celsius later in Simulink, before being sent to EnergyPlus.

3.2.2 Getting Votes from the Users

All votes are collected by the means of a user interface. In the ideal implementation discussed above, a user ID and password (as indicated in point 1 in Figure 3.2.2) would be used in order to identify and authenticate individual users. However, since a simple version is being implemented in this study, these parts are included conceptually, but not implemented in the user interface. Instead, five unique applications are used to ensure that there is a distinction between the different users and their votes.

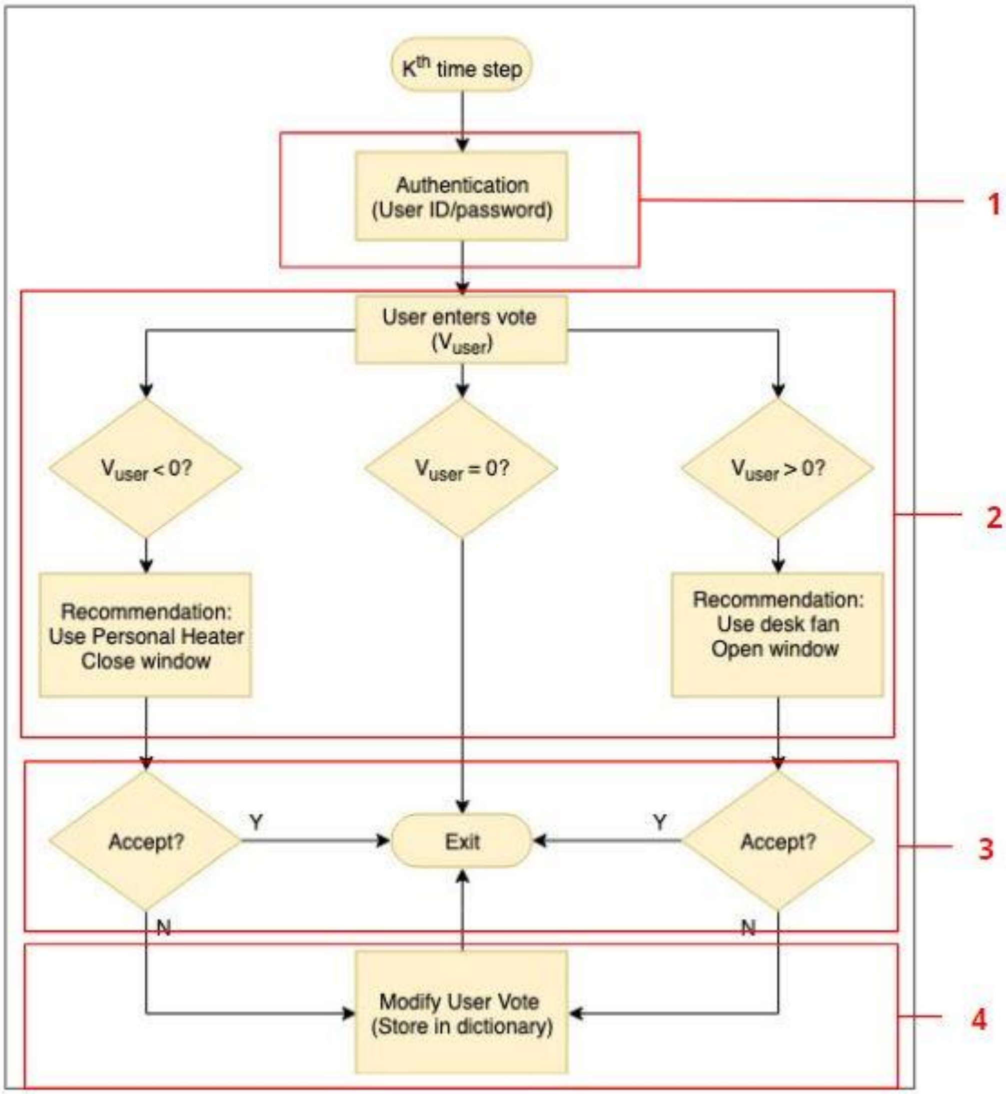


Figure 3.2.2 Schematic Diagram of Logic to Obtain Votes from Users

When the votes are obtained, the logic checks the value of the votes. If the values are negative, recommendations such as “turn personal heater on” or “put on a jacket” are provided, since negative values correspond to cold sensation. This is reflected in part 2, indicated in Figure 3.2.2. Similar logic can be ascribed to the positive votes as well.

In part 3 of the same figure, the user is then given the choice to accept or reject the recommendation. If the user accepts the recommendation, the user's vote is not updated in the database. If the user does not accept the recommendation, the stored value is updated, and used to send feedback to the controller when the next aggregation takes place.

The basic high level code for this part is as follows

- *@some time step k:*
- *Check User ID AND Password*
- *IF User is confirmed*
  - *Get vote from User*
  - *IF vote < 0*
    - *Provide recommendations to make user warmer using personal methods/devices*
  - *ELIF vote > 0*
    - *Provide recommendations to make user warmer using personal methods/devices*
  - *IF user accepts the vote:*
    - *EXIT*
  - *ELSE:*
    - *Update vote value in the database storing user votes*

Every 10 minutes, the values are reset to 0, and comfort is assumed unless the user expresses discomfort again.

3.2.3 Aggregating User Votes and Computing Temperature Setpoint Adjustment

Once the values or votes provided by the user are updated in the database, the values should be aggregated. There were some challenges in the process of aggregation and setpoint temperature adjustment, which are mentioned below:

1. When the scale has both negative and positive values, averaging the values would lead to a misrepresentation. Therefore, a different way of aggregation had to be devised.
2. The degree of modification to the temperature (ΔT) based on the aggregated value

The general concept followed was the rule of majority. The votes were separated based on whether they were hot or cold votes, since they are contrasting opinions. After they were separated into two separate lists, the number of users feeling hot or cold were compared, and majority opinion was prioritized. For example, if majority of the users felt hot, then the temperature setpoint would be increased ($\Delta T > 0$). In case the values turned out to be equal, the average of both the voting groups (that is, their standard deviation from 0) was compared. In this comparison, the group with a higher absolute average was considered, since a higher absolute average indicates a higher degree of discomfort. The general flowchart depicting this process is shown in Figure 3.2.3, and due to the elongated nature of this chart, the Figure is divided into two parts and enlarged in Figures 3.2.4 and 3.2.5.

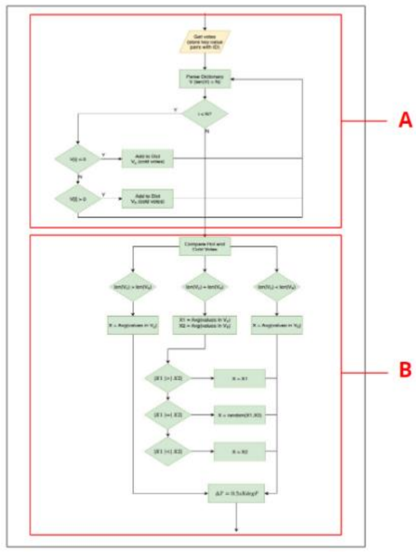


Figure 3.2.3 Key for aggregation logic, expanded in Figures 3.2.4 and 3.2.5

The pseudo code driving this logic is as follows:

- *@some time step k:*
- *WHILE there are User Vote Values:*
  - *IF Value < 0:*
    - *Append to list (VC) containing "Cold" votes (user feels cold)*
  - *ELIF Value > 0:*
    - *Append to list (VH) containing "Hot" votes (user feels hot)*
- *IF len(VC) > len(VH):* ***(Start of Part B of Figure 3.2.3)***
  - *len(VC) = NC*
  - $\bar{X} = (\sum_{i=1}^{NC} VC_i) / NC$
- *IF len(VC) < len(VH):*
  - *len(VH) = NH*

- $\overline{X} = (\sum_{i=1}^{NH} VH_i) / NH$

– *IF len(VC) == len(VH):*

- *len(VC) = len(VH) = N*
- $\overline{X_c} = (\sum_{i=1}^{N} VC_i) / N$
- $\overline{X_H} = (\sum_{i=1}^{N} VC_i) / N$
- *IF* $|\overline{X}_1| > |\overline{X}_2|$ *:*
  - $\overline{X} = \overline{X_1}$
- *ELIF* $|\overline{X}_1| < |\overline{X}_2|$ *:*
  - $\overline{X} = \overline{X_2}$
- *ELIF* $|\overline{X}_1| == |\overline{X}_2|$ *:*
  - $\overline{X} = \text{random}(\overline{X_1}, \overline{X_2})$

– $\Delta T = -(0.5 \times \overline{X})$ °F

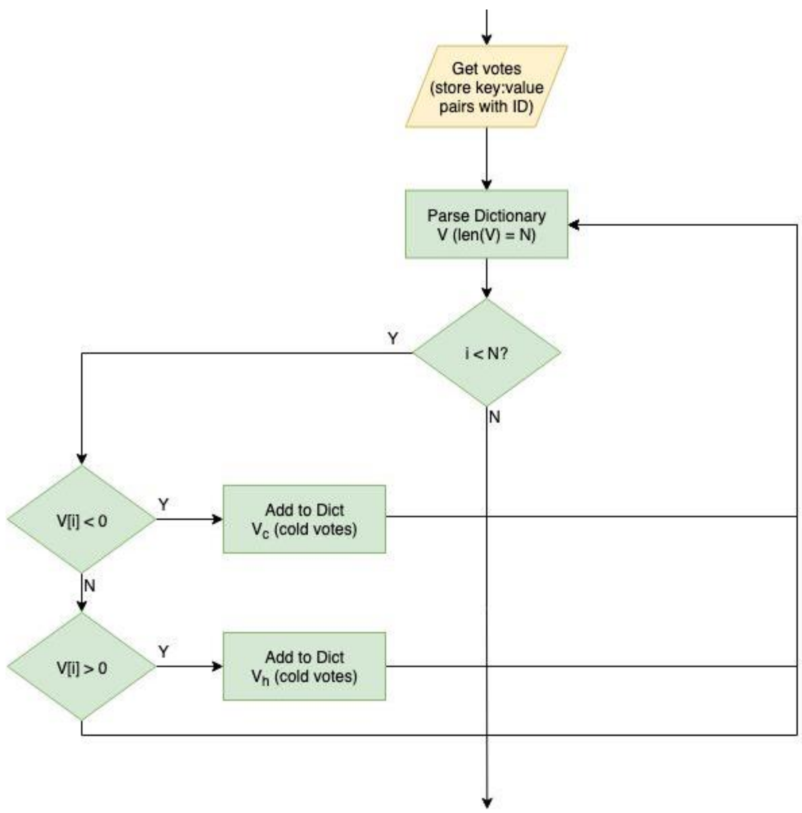


Figure 3.2.4 Part **A** of the schematic logic in Figure 3.2.3

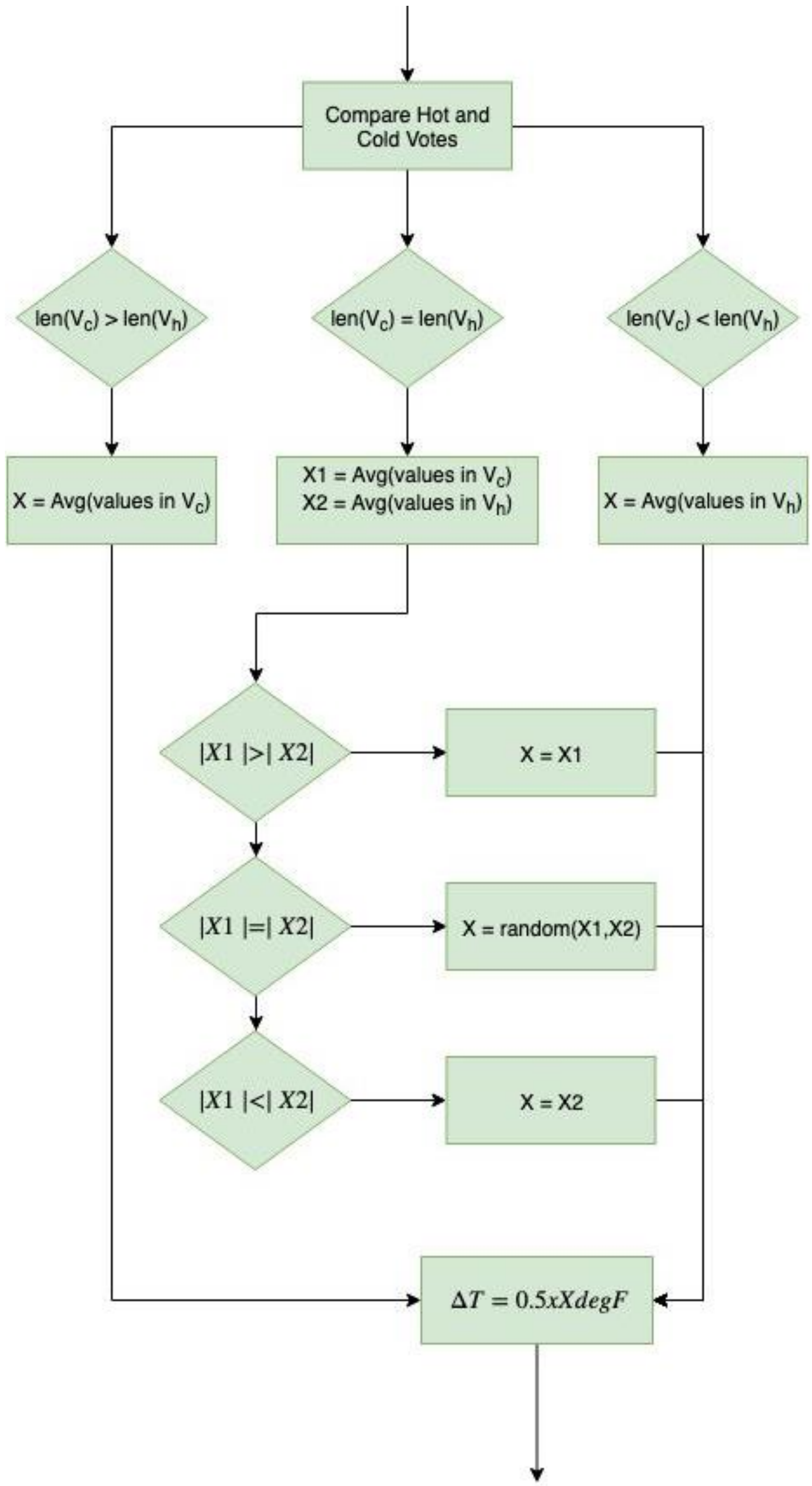


Figure 3.2.5 Part **A** of the schematic logic in Figure 3.2.3

4 User Interface Design

Two user interfaces were designed; One for the general occupants, and one for the facility managers. A distinction was made based on the type and degree of information needed for the occupants. The details of the interface design process for each of the two sub-groups is provided in sections 4.1 and 4.2.

4.1 User Interface for General Occupants

While designing the user interface for the occupants, the aim was to keep information minimal and display only strictly relevant information. Much like the thermostats in the existing building, the proposed design eliminates the display of current setpoint temperature. People sometimes tend to have preconceived notions attached to certain temperature setpoint. To eliminate this psychological impact, the temperature is not displayed in the interface for general occupants. Different components in the user interface are highlighted in Figure 4.1.1, and described in detail following the figure.

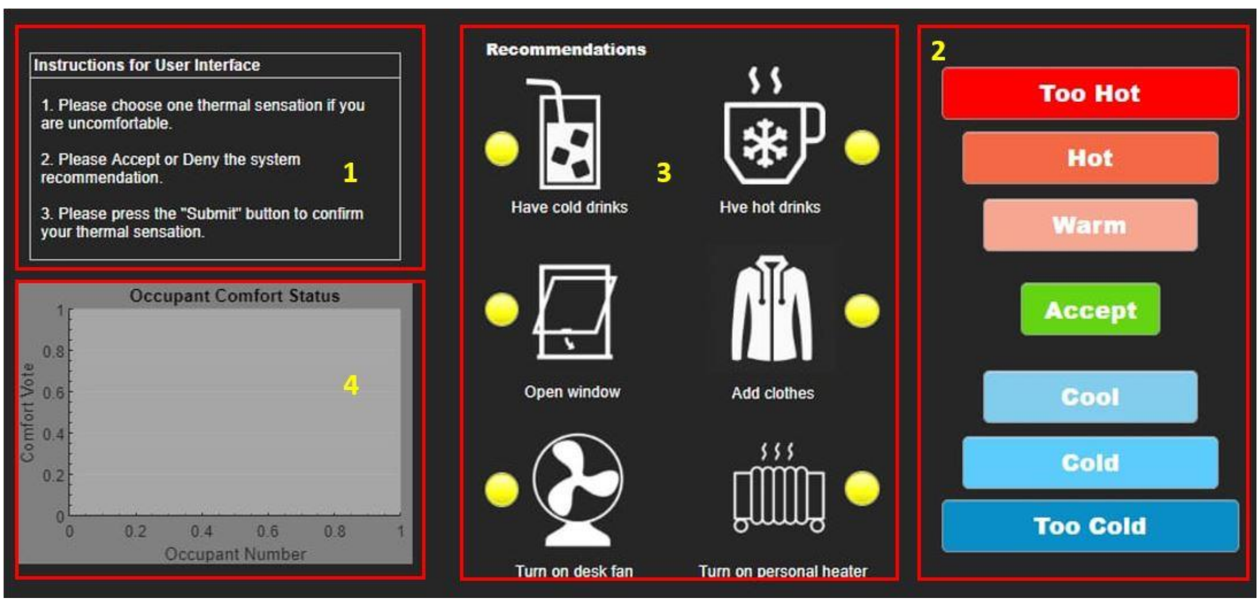


Figure 4.1.1 User Interface designed for General Occupants

1. **The Instruction Panel**: Instructs the user on how to use the application
2. **Thermal Vote Panel**: A 7 point scale was used to allow users to vote for a range of thermal sensations. Although it is anticipated that most users will tend to avoid the extremes, the extreme buttons (too hot and too cold) were retained in order to provide an extreme form of feedback which can notify the facility maintenance department of any serious malfunctions in the HVAC system. The colors were chosen to intuitively represent warm and cool sensations.

3. **Recommendations Panel**: This panel provides users simple suggestions based on whether they feel hot and cold, which the user may accept or deny. If the user chooses to accept the recommendations provided, the system assumes that the user does not need to vote to modify the thermal condition of the office, and their vote is disregarded.

4. **Occupant Comfort Status Panel**: This panel displays the comfort status of other occupants in the zone. Like the decision to exclude temperature setpoint from the interface, this is included for psychological reasons. Once the user provides their votes, they are provided with a set of recommendations as well as the comfort status of other occupants. It is possible that if the users were to see that most of the occupants sharing their zones were comfortable, the users might be more likely to accept a recommendation.

4.2 User Interface for Facility Managers

The facility managers need more detailed information about influencing parameters, such as the indoor and outdoor temperatures, zone setpoint temperatures, relative humidity etc. They are also provided with a real-time view of each occupant's comfort status, and can inspect anomalous votes by observing that occupant's microclimate if needed (e.g., if the occupant left a personal heater on by mistake, forgot about it, and now feels extremely hot). Figure 4.2.1 illustrates the different components in the Facility Manager's interface, and a detailed description of each component is provided following the Figure. All temperatures are in °F, to reflect the predominant practice in the United States.

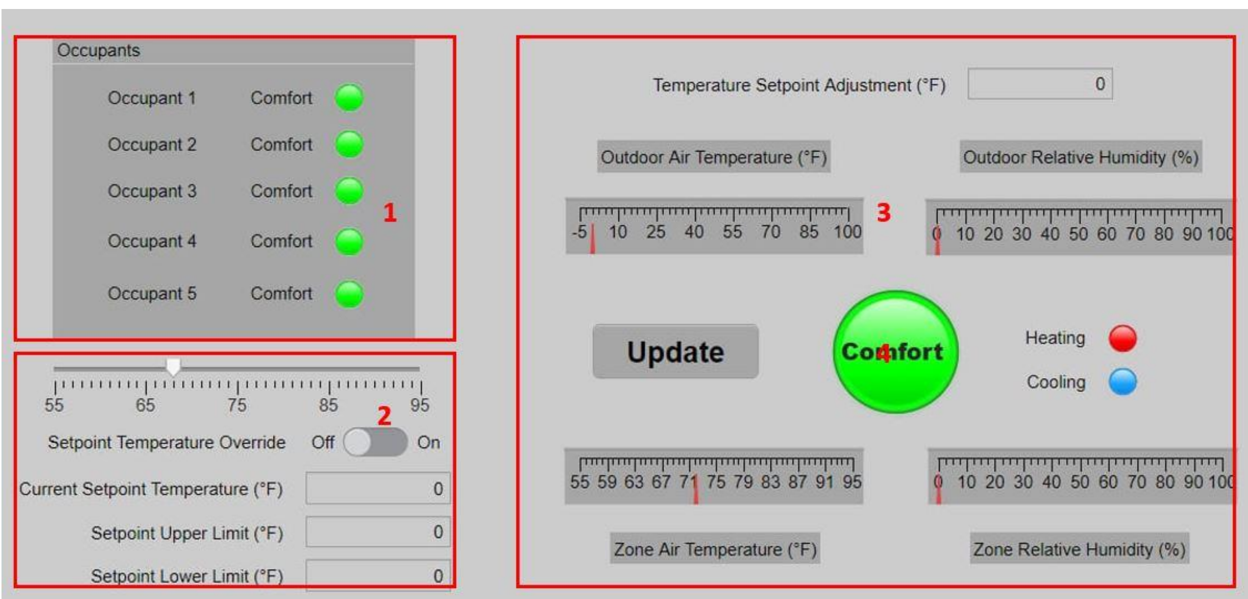


Figure 4.2.1 User Interface Designed for Facility Managers

1. **Comfort Status Panel**: This enables the facility manager to keep a check on the real time comfort status of individual occupants in a given zone. The color of the lights corresponds to the thermal vote of the users, matching with the occupant user interface.

2. **Override Panel**: The setpoint temperature switch toggles between on and off. When the switch is on, the managers can override the temperature setpoint arrived at by vote aggregation, and manually change it to what they deem appropriate for the situation. This button is provided in case the logic fails in some situation, as a fail-safe. Once the button is on, the slider above the button can be used to set the temperature. The “Current Setpoint Temperature” box displays the current zone temperature, regardless of the override status. The upper and lower limits for the setpoint temperature are also assigned here.

3. **Data Panel:** Factors influencing indoor thermal comfort and the ability of the system to meet its setpoint are provided in this panel. Although it would be ideal to utilize more specific weather data for Outdoor Air Temperature and Outdoor Relative Humidity, due to limitations of time and knowledge the results of an Energy Plus simulation were used for the demonstration period. The green “Comfort” button takes in the PMV output from the EnergyPlus simulation, and provides the facility manager information about the calculated comfort.

4. **Screen 2:** Screen 2, indicated in Figure 4.2.2., is used to display outdoor temperature and zone air temperature. Data is projected for 2 hours into the future, based on EnergyPlus simulation result.

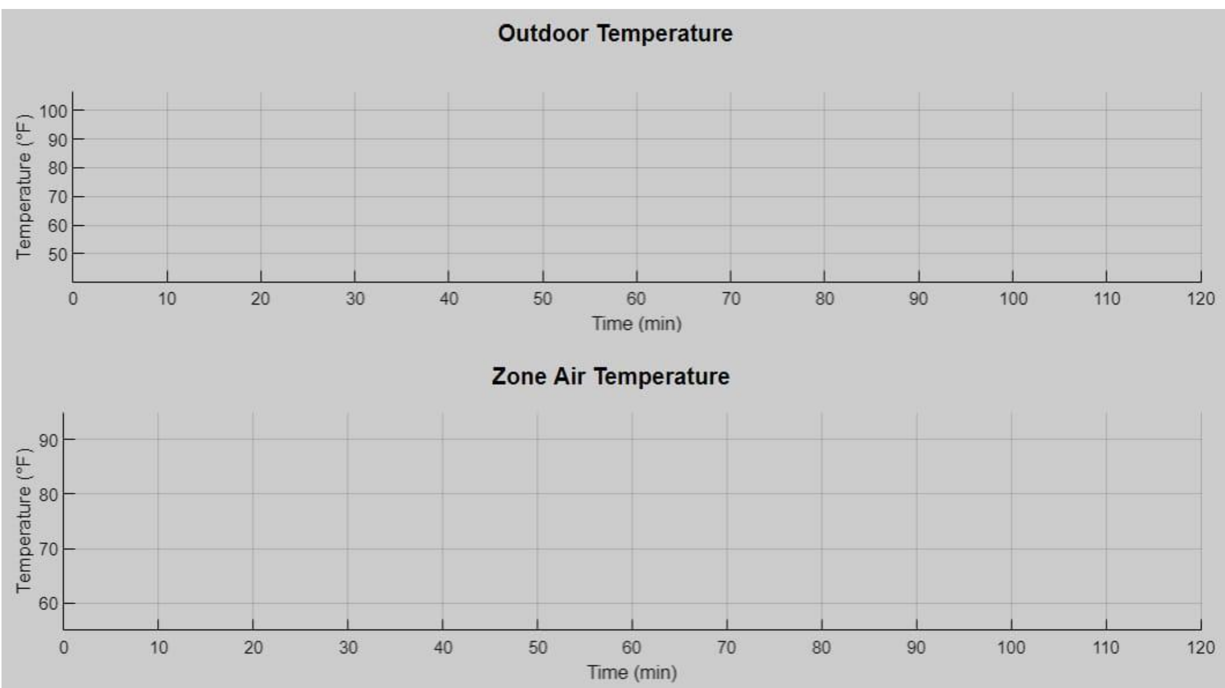


Figure 4.2.2 Screen 2 of the Facility Manager interface

5 Data Communication between the User Interface and Simulink

In order to enable occupants and facility managers to have control over heating and cooling systems in CSL building, we must connect the designed User Interface with Simulink model. The Matlab component we utilize to create user interface is App Designer. To facilitate the communication between User Interface and Simulink model, we first to tried to store data in the workspace in Matlab and control the Simulink model directly through User Interface. However, we couldn't figure it out as App Designer adopts a slightly different syntax, which resembles Matlab language but also contains something new. Thus, we tried a different way to connect UI and Simulink model, which decouples User Interface and Simulink model. In this method, UI will record the occupant votes in a local XLS file and read another CSV file recorded by the Simulink model to plot the current weather and thermal conditions. The Simulink model will read the occupants votes recorded by UI to calculate the setpoint temperature adjustment and then add it to current setpoint temperature to generate adjusted setpoint temperature for the EnergyPlus to run simulations. Then, Simulink Model will record the outputs (current outdoor and indoor temperature & relative humidity as well as future projections of these variables) from EnergyPlus in CSV files so that they can be read by the User Interface.

5.1 User Interface for Occupants

The occupant user interface contains two kinds of functions, one is the startup function, which will read the current comfort status of all occupants and plot them as a bar chart. Besides, it will also set all the recommendation lights back to default color (white). The example code is shown in figure 5.1.1. The other function is the button-push function, which will substitute the original value with another value (-3 ~ 3) in the XLS file based on the button pushed, update the bar chart that displays occupant comfort status, and change the related recommendation light from white to green and restore all other lights into default (white). The example code for too-hot button is shown in figure 5.1.2.

```
function startupFcn(app)

    n1 = xlsread('vote.xls','Sheet1','A2');
    n2 = xlsread('vote.xls','Sheet1','B2');
    n3 = xlsread('vote.xls','Sheet1','C2');
    n4 = xlsread('vote.xls','Sheet1','D2');
    n5 = xlsread('vote.xls','Sheet1','E2');
    y = [n1 n2 n3 n4 n5];
    bar(app.UIAxes,y);

    app.Lamp.Color = 'White';
    app.Lamp_2.Color = 'White';
    app.Lamp_3.Color = 'White';
    app.Lamp_4.Color = 'White';
    app.Lamp_5.Color = 'White';
    app.Lamp_6.Color = 'White';
end
```

Figure 5.1.1 Screenshot of startup function

```
function TooHotButtonPushed(app, event)
    app.Lamp.Color = 'White';
    app.Lamp_2.Color = 'White';
    app.Lamp_3.Color = 'Green';
    app.Lamp_4.Color = 'White';
    app.Lamp_5.Color = 'White';
    app.Lamp_6.Color = 'White';

    xlswrite('vote.xls',3,'Sheet1','A2');

    n1 = xlsread('vote.xls','Sheet1','A2');
    n2 = xlsread('vote.xls','Sheet1','B2');
    n3 = xlsread('vote.xls','Sheet1','C2');
    n4 = xlsread('vote.xls','Sheet1','D2');
    n5 = xlsread('vote.xls','Sheet1','E2');
    y = [n1 n2 n3 n4 n5];
    bar(app.UIAxes,y);
end
```

Figure 5.1.2 Screenshot of button-push function for too-hot button

5.2 User Interface for Facility Managers

The facility manager user interface contains three kinds of functions, one is the startup function, which can will read local files to display comfort status for each occupant, current setpoint temperature, setpoint temperature adjustment and weather & thermal conditions as well as plotting timeseries data regarding to outdoor and indoor air temperature. It will also display on which mode is the system operating (heating or cooling), and overall comfort status based on PMV generated by EnergyPlus. The example code is shown in figure 5.2.1 and figure 5.2.2. One is the switch function, which can be used by facility manager to override current occupant votes. It will record current control status and current number on sliders to XLS file and number '1' means override is on, while number '0' means override is off and occupants can change the setpoint temperature by voting. The example code is shown in figure 5.2.3. Another is the button-push function, which will update all the information mentioned in startup function. The code is the same as startup function.

```
function startupFcn(app)
    app.HeatingLamp.Color = 'White';
    app.CoolingLamp.Color = 'White';

    app.TemperatureSetpointAdjustmentFEditField.Value = xlsread('adjust_ST.xls','Sheet1','A2');
    app.CurrentSetpointTemperatureFEditField.Value = xlsread('adjust_ST.xls','Sheet1','C2');

    p = csvread('CoSimResult.csv');
    a = p(end-287:end,:);

    app.OutdoorAirTemperatureFGauge.Value = a(1,1);
    app.OutdoorRelativeHumidityGauge.Value = a(1,2);
    app.ZoneAirTemperatureFGauge.Value = a(1,5);
    app.ZoneRelativeHumidityGauge.Value = a(1,6);
    PMV = a(1,11);

    t1 = a(1:13,1);
    z1 = transpose(t1);
    y1 = z1*1.8+32;
    x = 0:10:120;
    line(app.UIAxes_2,x,y1)

    t2 = a(1:13,5);
    z2 = transpose(t2);
    y2 = z2*1.8+32;
    x = 0:10:120;
    line(app.UIAxes_3,x,y2)
```

Figure 5.2.1 Screenshot of displaying current weather & thermal conditions and timeseries data

```
if -0.5 <= PMV <= 0.5
    app.ComfortLamp.Color = 'Green';
else
    app.ComfortLamp.Color = 'Red';
end
if app.OutdoorAirTemperatureFGauge.Value - app.ZoneAirTemperatureFGauge.Value > 9
    app.CoolingLamp.Color = 'Blue';
elseif app.OutdoorAirTemperatureFGauge.Value - app.ZoneAirTemperatureFGauge.Value < -9
    app.HeatingLamp.Color = 'Red';
else
    app.CoolingLamp.Color = 'White';
    app.HeatingLamp.Color = 'White';
end
n1 = xlsread('vote.xls','Sheet1','A2');
if n1 == 3
    app.ComfortLamp_2.Color = [1 0 0];
elseif n1 == 2
    app.ComfortLamp_2.Color = [0.9608 0.4078 0.2706];
elseif n1 == 1
    app.ComfortLamp_2.Color = [0.9686 0.651 0.5725];
elseif n1 == -1
    app.ComfortLamp_2.Color = [0.5098 0.8039 0.9294];
elseif n1 == -2
    app.ComfortLamp_2.Color = [0.3569 0.8 0.9882];
elseif n1 == -3
    app.ComfortLamp_2.Color = [0.0392 0.5569 0.7804];
else
    app.ComfortLamp_2.Color = [0.3922 0.8314 0.0745];
end
```

Figure 5.2.2 Screenshot of showing comfort status (individual and overall) and operation mode

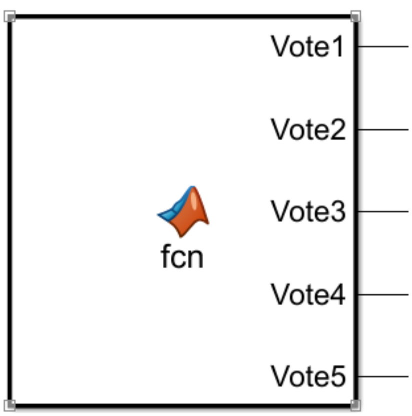
```
function SwitchValueChanged(app, event)
    value = app.Switch.Value;
    if isequal(value,'On')
        x = 1;
        xlswrite('adjust_ST.xls',app.SetpointTemperatureOverrideSlider.Value,'Sheet1','D2');
    else
        x = 0;
    end
    xlswrite('adjust_ST.xls',x,'Sheet1','B2');
end
```

Figure 5.2.3 Screenshot of switch function

5.3 Simulink Model

The Simulink model needs to do the following things to run in the intended way and support data display in user interface.

1) Read occupant votes and restore votes to default value so that it won't influence future votes. The example code is shown in figure 5.3.1.

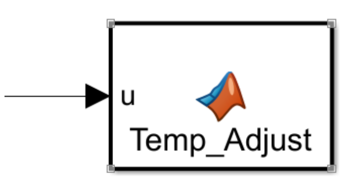


Read vote data from a xls file

```
function [Vote1,Vote2,Vote3,Vote4,Vote5]= fcn
coder.extrinsic('xlsread');
coder.extrinsic('xlswrite');
Vote1 = 0;
Vote2 = 0;
Vote3 = 0;
Vote4 = 0;
Vote5 = 0;
Vote = xlsread('vote.xls');
Vote1 = Vote(1,1);
Vote2 = Vote(1,2);
Vote3 = Vote(1,3);
Vote4 = Vote(1,4);
Vote5 = Vote(1,5);
xlswrite('vote.xls',0,'Sheet1','A2');
xlswrite('vote.xls',0,'Sheet1','B2');
xlswrite('vote.xls',0,'Sheet1','C2');
xlswrite('vote.xls',0,'Sheet1','D2');
xlswrite('vote.xls',0,'Sheet1','E2');
```

Figure 5.3.1 Code for reading occupant votes

2) Record setpoint temperature adjustment in XLS file so that it can be displayed in facility manager user interface. The example code is shown in figure 5.3.2.

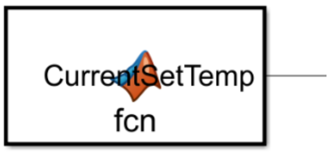


Export cooing setpoint temp adjustment to a xls.file

```
function Temp_Adjust(u)
%#codegen
coder.extrinsic('xlswrite');
xlswrite('adjust_ST.xls',u,'Sheet1','A2');
```

Figure 5.3.2 Code for recording setpoint temperature adjustment

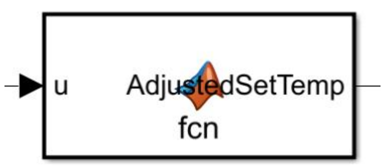
3) Read current setpoint temperature and add setpoint temperature adjustment to it to obtain the adjusted setpoint temperature for EnergyPlus simulation. The example code is shown in figure 5.3.3.




```
function CurrentSetTemp  = fcn
CurrentSetTemp = 0;
coder.extrinsic('xlsread');
CurrentSetTemp = xlsread('adjust_ST.xls','Sheet1','C2');
```

Figure 5.3.3 Code for reading current setpoint temperature

4) Record adjusted setpoint temperature in XLS file as it is current setpoint temperature for next time step. In this process, if function is utilized to make the judgment the control status (whether override is on or off) and record it based on control status. The example code is shown in figure 5.3.4.



Calculate adjusted cooling setpoint temp and export it to a xls.file

```
function AdjustedSetTemp  = fcn(u)
%#codegen
coder.extrinsic('xlswrite');
coder.extrinsic('xlsread');
a = 0;
AdjustedSetTemp = 0;
a = xlsread('adjust_ST.xls','Sheet1','B2');
if a == 0
    AdjustedSetTemp = u;
else
    AdjustedSetTemp = xlsread('adjust_ST.xls','Sheet1','D2');
end
xlswrite('adjust_ST.xls',AdjustedSetTemp,'Sheet1','C2');
```

Figure 5.3.4 Code for recording adjusted setpoint temperature

6. Interaction of Matlab, Simulink and EnergyPlus

6.1 Overview

Similar to a web application development, the User Interface (UI) is developed as a frontend in the Matlab App Designer to interact with occupants and facility manager, then control logics are coded in as backend in Simulink to use the data from UI as one of the inputs to calculate temperature setpoint adjustment, and output an adjusted temperature setpoint. The EnergyPlus is configured as a simulation engine, using Building Controls Virtual Test Bed (BCVTB) as the simulation environment, which allows to co-simulate a building in EnergyPlus and the HVAC and control system in Modelica. It takes the adjusted temperature setpoint as inputs, and outputs zone dry-bulb air temperature, relative humidity, and Predicted Mean Vote, etc. These simulation results are sent back to Simulink as inputs for control logic, located inside the controller shown below. Together with inputs data from UI, the co-simulation process can be iterated until the end.

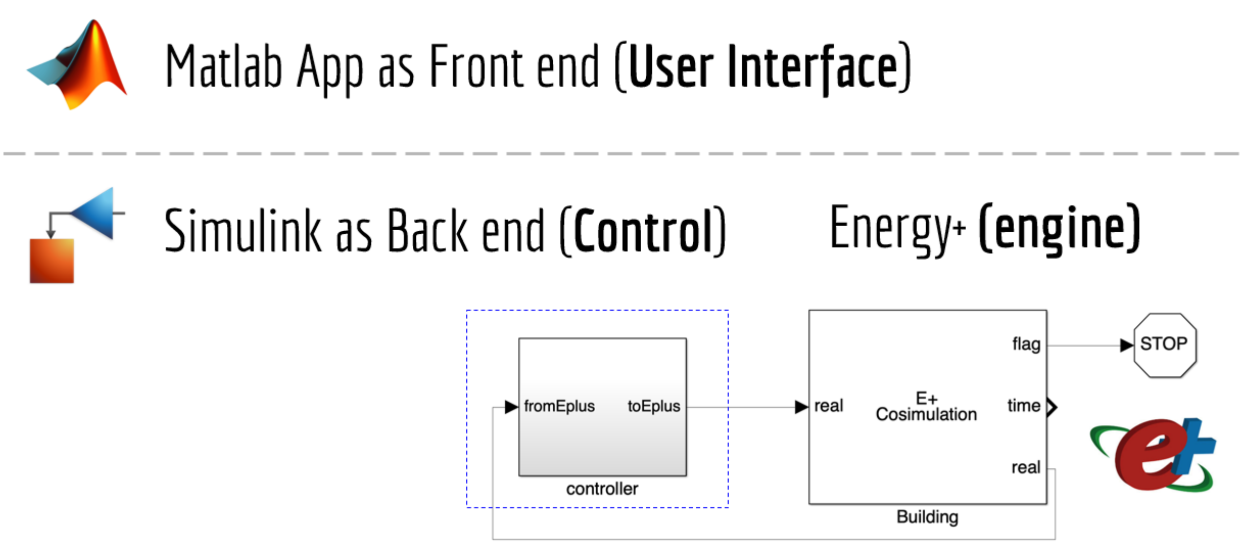


Figure 6.1.1 Interaction between Matlab, Simulink and EnergyPlus

6.2 Temperature degree adjustment control logic

Votes from occupants are recorded in a XLS file, by coding in the user interface in Matlab, then in Simulink, the control logic reads the file and take the votes data as inputs. Then in the control logic, total hot or cold vote value and vote number are counted respectively. For example, if the number of hot votes is larger than the number cold votes, than the decision would be cooling down the temperature setpoint. The adjustment is an average value, which is the total hot vote value over total hot vote number. If the number of hot votes is smaller than the number of cold votes, than the decision would be heating up the temperature setpoint. The adjustment is the total cold vote value over total cold vote number. If the number of hot votes is equal to the number of cold votes, than the decision depends on the total value of hot and cold votes. The adjustment is the total value over total cold or hot vote number. Of particular note that there is a switch block to avoid the situation of zero as the divisor, as when all occupants vote hot or cold, or all occupants feel comfortable, the number of hot or cold vote would be zero. Finally, according to the literature review in the previous chapter, the 0.5 degree adjustment for temperature is reasonable for every one vote difference.

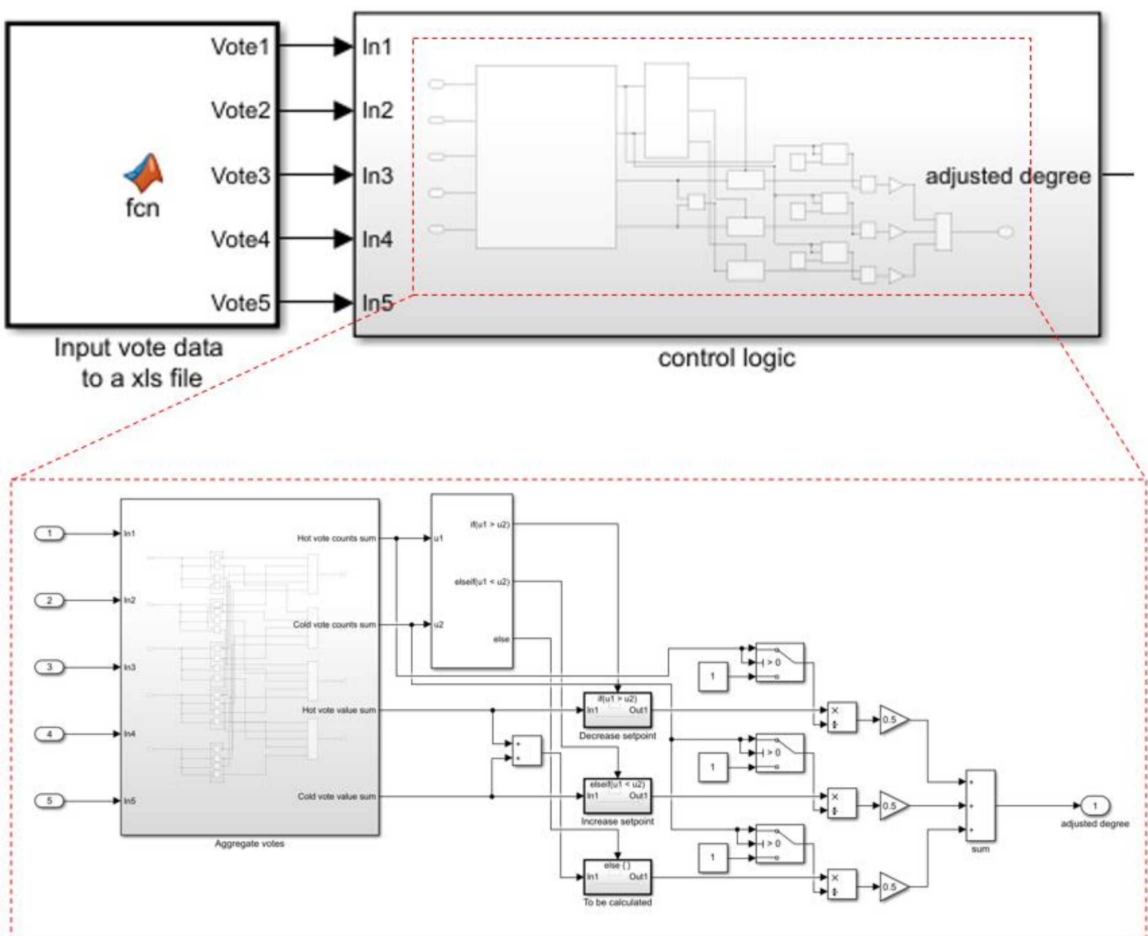


Figure 6.2.1 Vote data form user interface as input to control logic

In detail, the counting of total hot or cold vote number and value is through judging the sign of vote. For counting the number of the hot votes, if the sign of a vote is positive, than judging output would be 1, which is added to the hot vote counts. For counting the number of the cold votes, if the sign of vote is negative or 0, than judging output would be 1, which is added to the cold vote counts. For summing the value of all hot votes, if the vote is positive, then output would be the hot judging value times the value, which is 1 times the value. So the value itself would be added to the total hot value block. If the vote is negative, then output would be 0 times the value, which is 0 and would not influence the total hot value block. For summing the value of all cold votes, if the vote is negative, then output would be the cold judging value times the value, which is 1 times the value. So the value itself would be added to the total cold value block. If the vote is negative, then output would be 0 times the value, which would not influence the total hot value block.

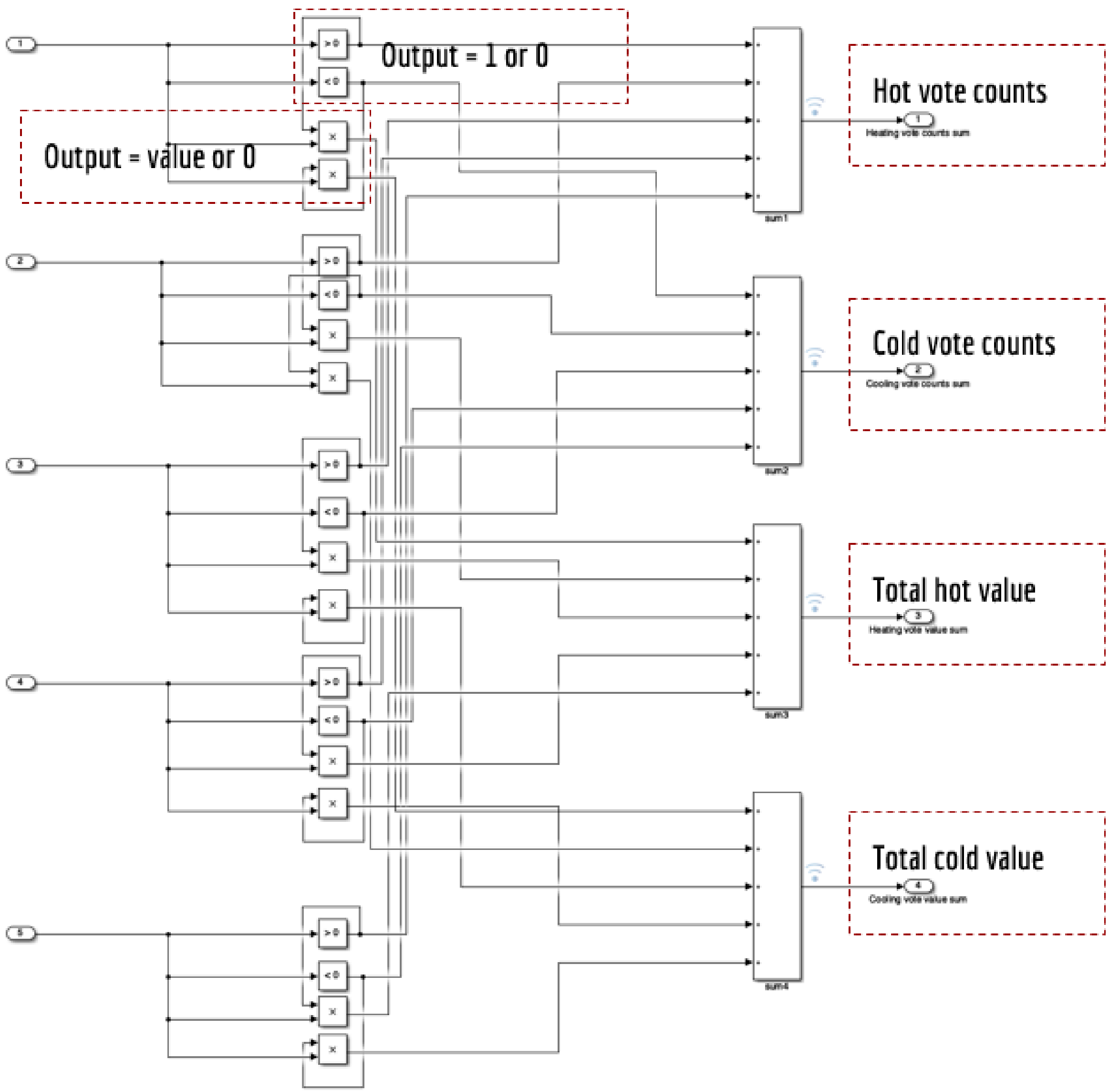


Figure 6.2.2 Vote number counting and value summing logic

6.3 Temperature setpoint as inputs to user interface and EnergyPlus

The adjusted degree is recorded in cell A2 in a XLS file, named as 'adjust_ST'. The current temperature setpoint data is from user interface in Matlab, which is actually the temperature setpoint adjusted at last time step, recorded in cell C2 in the same file. At the beginning of the simulation, it is a default value at 75. Then in Simulink, the current temperature setpoint data is read and is added by the adjusted degree, outputting adjusted cooling setpoint. Of particular note that the 'adjust_ST' file also recorded temperature setpoint in cell D2, which is inputted by facility manager in the user interface in Matlab. If the facility manager decided to control the setpoint, then it would be recorded as 1 in cell B2 in the same file. In Simulink, if the control status is 1, then the cooling setpoint inputted by facility manager will override the cooling setpoint generated by the vote control logic. Also, facility manager can set lower limit and upper limit to the setpoint adjustment range, recorded in cell E2 and F2. Finally, through transforming from Fahrenheit to Celsius, together with default heating setpoint, the adjusted cooling setpoint is sent to EnergyPlus.

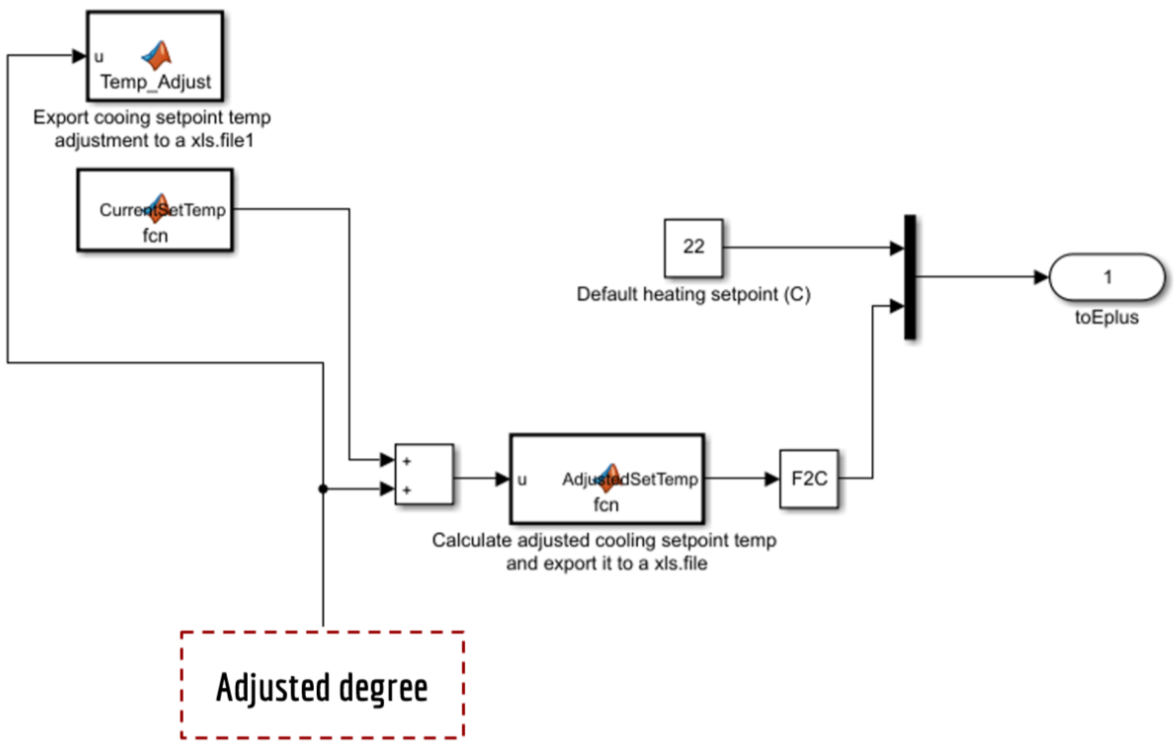


Figure 6.3.1 Adjusted degree as input to user interface and EnergyPlus

	A	B	C	D	E	F
1	Tempature Adjustment	Control Status	Current Set Temp	Override Set Temp	Set Temp lower limit	Set Temp upprt limit
2	-1.5	0	75	70	68	82

Figure 6.3.2 Data recorded in the adjust_ST.xls file

6.4 EnergyPlus, MLE+ and BCVTB configuration

First of all, the MLE+ is installed in Matlab, and can be connected as a toolbox in Simulink. It will configure paths of working files and programs: EnergyPlus executable file, which is called "RunEPlus.bat"; IDF file of the building; weather file for weather data input; the folder in which the Simulink program located; BCVTB executable file. In addition, the whole simulation time is 1 week, which is about 600000 seconds. Each time step is 10 minutes, which is 600 seconds. In terms of outputs, total 11 output variables are added in the IDF file of CSL and the corresponding variables configuration file, created by PhD Zhiang. They are respectively outdoor air temperature, outdoor relative humidity, diffuse solar radiation, direct solar radiation, indoor air temperature, indoor relative humidity, mean radiant temperature, current cooling setpoint, supply air flow, supply air temperature and PMV value of that zone.

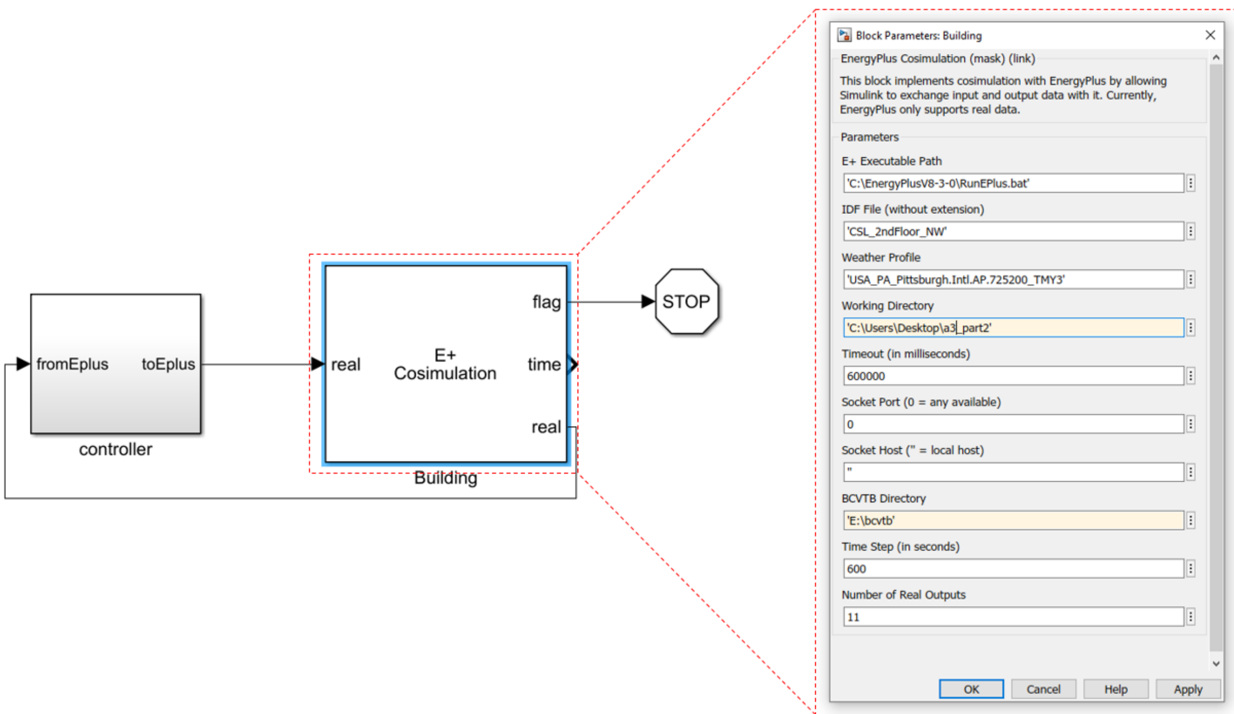


Figure 6.4.1 EnergyPlus, MLE+ and BCVTB configuration

```
<!-- Recieve from E+ -->
  <!-- Weather -->
  <variable source="EnergyPlus">
   <EnergyPlus name="Environment" type="Site Outdoor Air Drybulb Temperature"/>
  </variable>

  <variable source="EnergyPlus">
   <EnergyPlus name="Environment" type="Site Outdoor Air Relative Humidity"/>
  </variable>

  <variable source="EnergyPlus">
   <EnergyPlus name="Environment" type="Site Diffuse Solar Radiation Rate per Area"/>
  </variable>

  <variable source="EnergyPlus">
   <EnergyPlus name="Environment" type="Site Direct Solar Radiation Rate per Area"/>
  </variable>

  <!-- IAT all zones -->

  <variable source="EnergyPlus">
   <EnergyPlus name="2ND%FLOOR%SOUTH:2nd%Open%Office%NW" type="Zone Air Temperature"/>
  </variable>

  <variable source="EnergyPlus">
   <EnergyPlus name="2ND%FLOOR%SOUTH:2nd%Open%Office%NW" type="Zone Air Relative Humidity"/>
  </variable>

  <variable source="EnergyPlus">
   <EnergyPlus name="People 2ND%FLOOR%SOUTH:2nd%Open%Office%NW" type="Zone Thermal Comfort Mean Radiant Temperature"/>
  </variable>

  <!-- Cooling setpoint of all zones -->

  <variable source="EnergyPlus">
   <EnergyPlus name="2ND%FLOOR%SOUTH:2nd%Open%Office%NW" type="Zone Thermostat Cooling Setpoint Temperature"/>
  </variable>

  <!-- HVAC Energy -->
  <variable source="EnergyPlus">
   <EnergyPlus name="2ND%FLOOR%SOUTH:2nd%Open%Office%NW Single Duct VAV VSF Supply Inlet" type="System Node Mass Flow Rate"/>
  </variable>

  <variable source="EnergyPlus">
   <EnergyPlus name="2ND%FLOOR%SOUTH:2nd%Open%Office%NW Single Duct VAV VSF Supply Inlet" type="System Node Temperature"/>
  </variable>


  <!-- Occupant_PMV -->
  <variable source="EnergyPlus">
   <EnergyPlus name="People 2ND%FLOOR%SOUTH:2nd%Open%Office%NW" type="Zone Thermal Comfort Fanger Model Pmv"/>
  </variable>


<!-- Send to E+ -->
 <variable source="Ptolemy">
   <EnergyPlus schedule="ZoneHeatingSetpoint_2ndFloor2ndOpenNW"/>
 </variable>

 <variable source="Ptolemy">
   <EnergyPlus schedule="ZoneCoolingSetpoint_2ndFloor2ndOpenNW"/>
 </variable>

</BCVTB-variables>
```

Figure 6.4.2 Output variables configuration file

6.5 Assumption of PMV/PPD-based occupant votes prediction.

The total 11 output variables from EnergyPlus become inputs to the control logic, recorded in a XLS file, which is read and displayed by the user interface in Matlab. Here in the control logic, the last option of the inputs, which represents PMV value, is selected and judged. If the PMV value is less than -3, it means no occupancy at the zone. Then the adjusted degree will be 4 degree, as the default cooling setback is 28 degrees, based on the default cooling setpoint at 24 degrees, shown as Celsius below (Fahrenheit in real program). Otherwise, predicted votes will be generated by number of dissatisfied people, which is calculated by the function between PMV and percentage of predicted dissatisfaction (PPD). For example, if the absolute value of PMV is larger than 0.5 and less than 1, then corresponding PPD would be larger than 10% and smaller than 30%, which means that among 5 occupants, the dissatisfied people would be 1. Finally, random integers between -3 and 3 will be applied to corresponding dissatisfied people as votes.

Table 6.5.1 Dissatisfied people prediction based on PMV/PPD

abs (PMV)	PPD	Dissatisfied People (Among 5 occupants)	
[0.5, 1)	[10%, 30%)	[0.5, 1.5)	1
[1, 1.5)	[30%, 50%)	[1.5, 2.5)	2
[1.5, 2)	[50%, 70%)	[2.5, 3.5)	3
[2, 2.5)	[70%, 90%)	[3.5, 4.5)	4

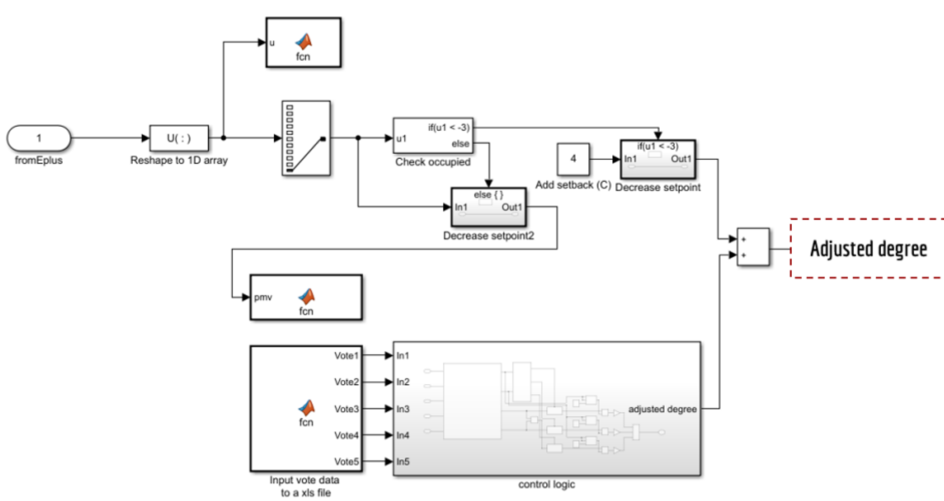


Figure 6.5.1 PMV/PPD-based votes prediction logic

```
function fcn(pmv)
coder.extrinsic('xlswrite')
vote1 = 0;
vote2 = 0;
vote3 = 0;
vote4 = 0;
vote5 = 0;
apmv = abs(pmv);

if apmv > 0.5
    if apmv < 1
        vote1 = randi([-3,3]);
    elseif apmv < 1.5
        vote1 = randi([-3,3]);
        vote2 = randi([-3,3]);
    elseif apmv < 2
        vote1 = randi([-3,3]);
        vote2 = randi([-3,3]);
        vote3 = randi([-3,3]);
    elseif apmv < 2.5
        vote1 = randi([-3,3]);
        vote2 = randi([-3,3]);
        vote3 = randi([-3,3]);
        vote4 = randi([-3,3]);
    end
end

xlswrite('vote.xls',vote1,'Sheet1','A2');
xlswrite('vote.xls',vote2,'Sheet1','B2');
xlswrite('vote.xls',vote3,'Sheet1','C2');
xlswrite('vote.xls',vote4,'Sheet1','D2');
xlswrite('vote.xls',vote5,'Sheet1','E2');
```

Figure 6.5.2 Matlab code for PMV/PPD-based vote prediction

7 Simulation Result Analysis

7.1 Overview

In order to analyze the performance for our proposed control logic, we did the co-simulation for a two-day period in August to analyze the comfort status in CSL building. We choose a two-day period because if the co-simulation period is too long, it takes much more time to run the simulation. Also, the accuracy of simulation results will be compromised as uncertainty will increase as simulation period increases. In order to test whether our control logic actually works or not, we should choose the time period when the cooling system is actually running. Thus, we choose the hottest two days in August, which is Aug 4th and Aug 5th, as shown in figure 7.1.1.

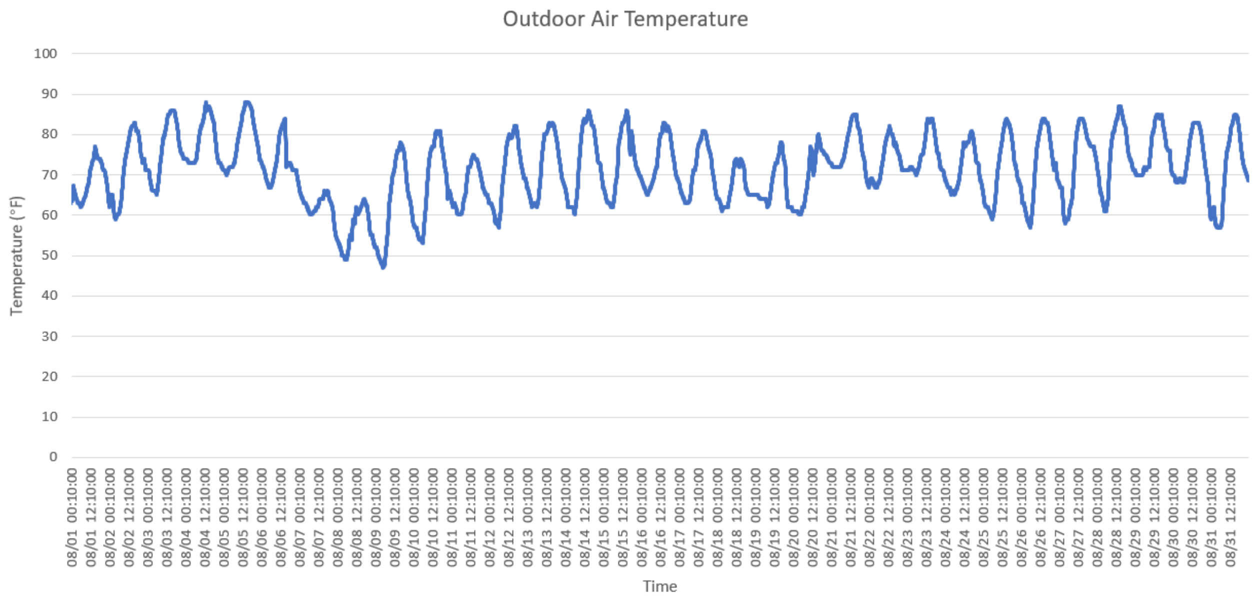


Figure 7.1.1 Outdoor air temperature in August for CSL building

7.2 Baseline (75 °F fixed setpoint temperature)

As shown in figure 7.2.1 we can notice that when setpoint temperature is fixed as 75°F, during the occupied time period, PMV is 100% within the comfort range (-0.5 ~ 0.5), with the average as -0.32548. However, PMV is always lower than 0, which means occupants will generally feel a bit chiller when they are working, which is consistent with the results of post occupancy evaluation (POE). We also want to consider energy consumption of northwestern open office area on the second floor, however, we only have the whole building energy consumption data. Besides, what we focus on is occupant thermal comfort, thus we don't include analysis of energy consumption in that zone.

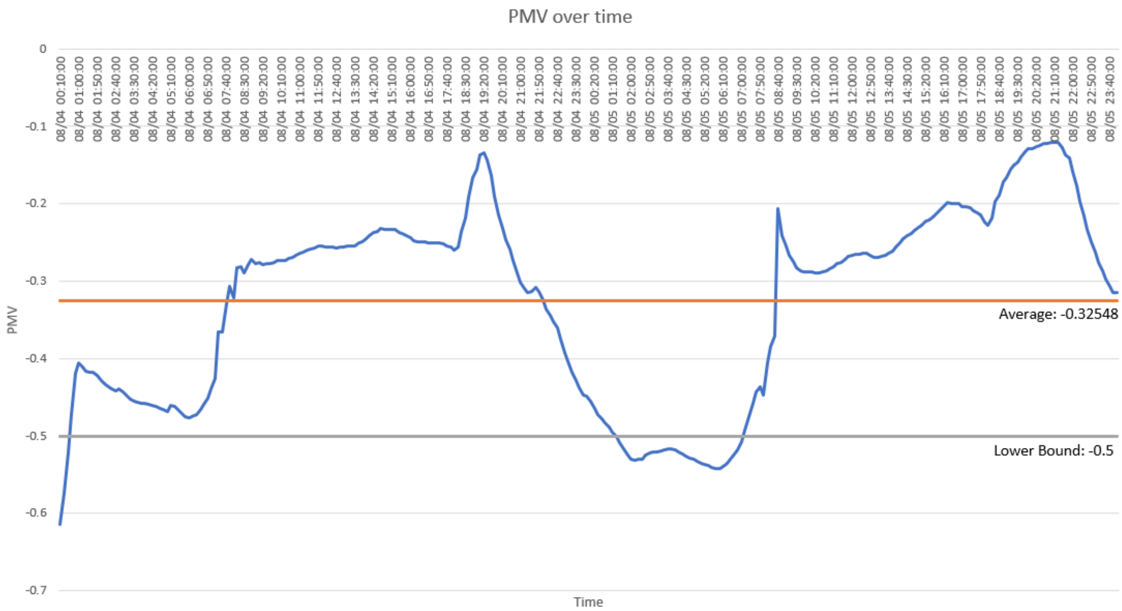


Figure 7.2.1 PMV over time for 75°F fixed setpoint temperature

7.3 Proposition (Occupant Voting-based HVAC Control)

As shown in figure 7.3.1, we can notice that when implementing the voting-based HVAC control logic, during the occupied time period, PMV is 100% within the comfort range (-0.5 ~ 0.5), with the average as -0.21615. Besides, during the occupied time period, the average PMV is close to 0, which is ideal for occupants.

However, when we dig deeper in the time series data, as shown in figure 7.3.2, we find that cooling setpoint temperature is always higher than zone air temperature, which means that cooling is actually always off during the simulation period. This is reasonable as CSL building has a high performing envelope that can maintain indoor thermal comfort without switching on HVAC system. And when HVAC system is off, no matter however you change the setpoint temperature, the indoor thermal condition will not be influenced. In this logic, there exists a tiny problem, which is that even the HVAC is off, setpoint temperature will change according to PMV. Thus, in order to solve this, we enable facility manager to enter reasonable upper and lower bounds for setpoint temperature.

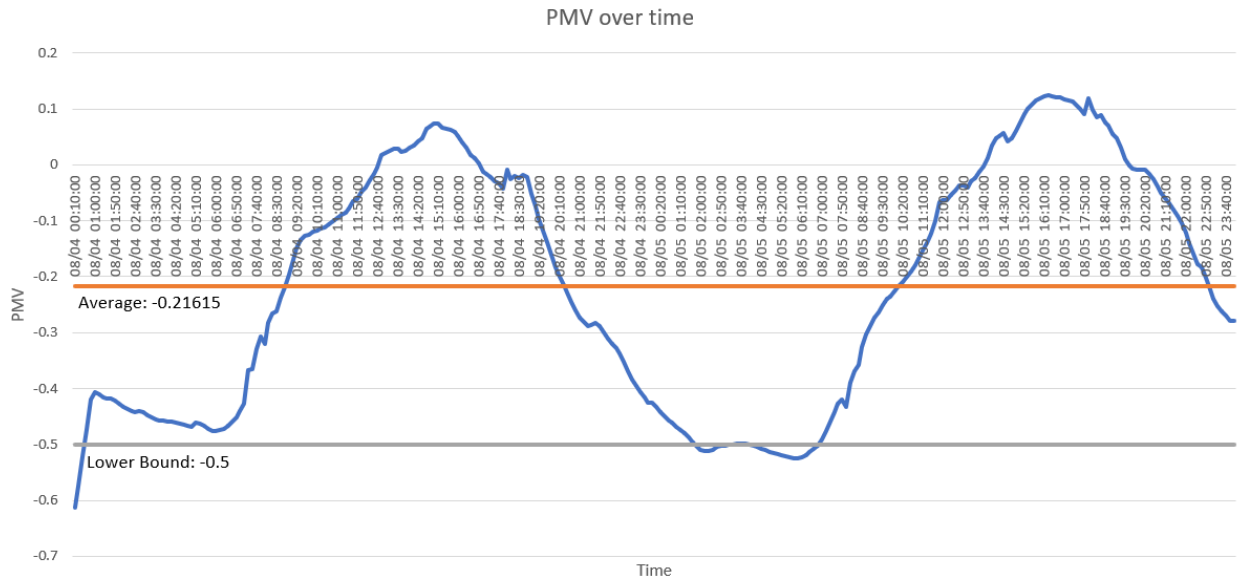


Figure 7.3.1 PMV over time for voting-based HVAC control logic

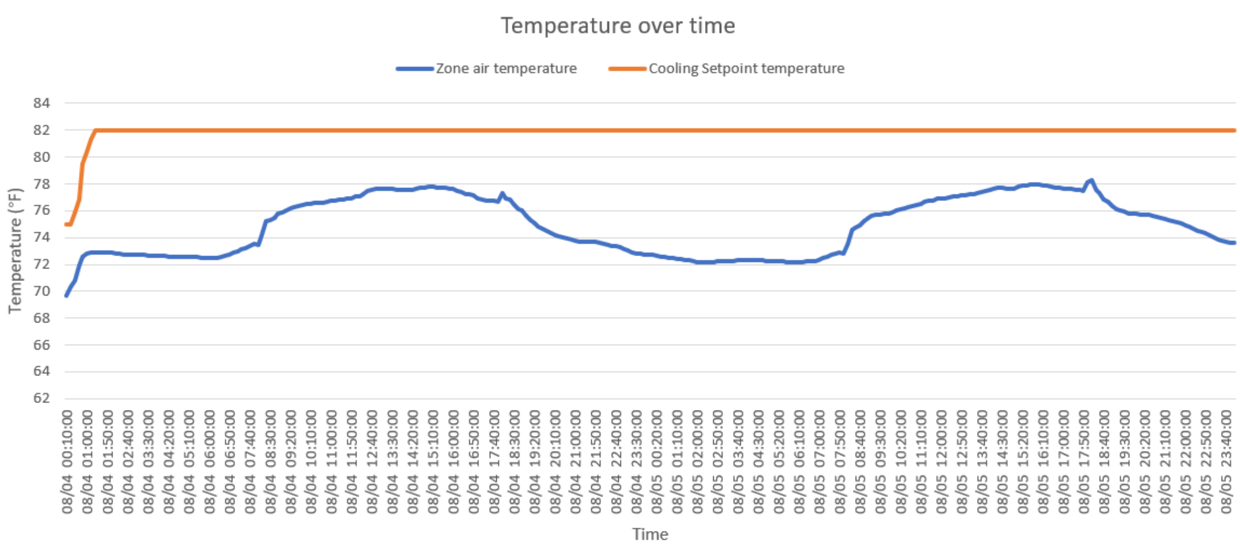


Figure 7.3.2 Zone air & setpoint temp over time for voting-based HVAC control logic

Compare PMV curve (figure 7.3.1) and outdoor air temperature curve (figure 7.3.3), we can notice that indoor thermal condition is actually influenced by outdoor air temperature as HVAC system is off.

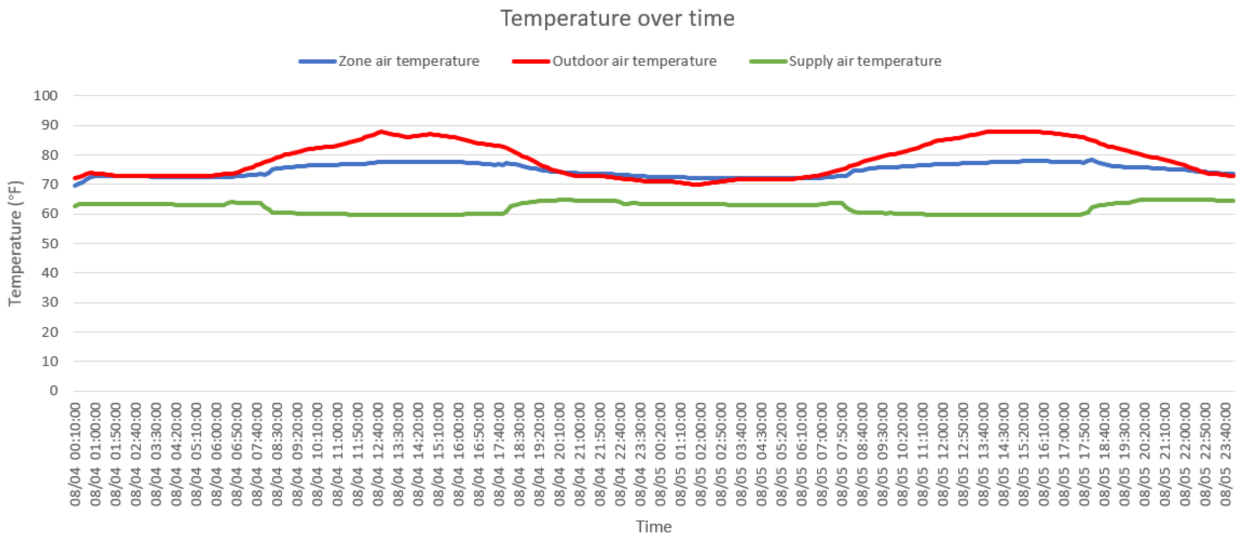


Figure 7.3.3 Zone, outdoor and supply air temperature over time for voting-based HVAC control logic

In addition, according to figure 7.3.3 and figure 7.3.4, we notice that even when zone air temperature change is smaller than outdoor air temperature change, this is partially due to the high performing envelope. Another important reason is that in order to meet ventilation needs during the occupied period, about 0.105 kg of fresh cool air (60°F ~ 61°F) should be supplied to CSL building per second. This help maintain zone air temperature between 76°F and 79°F, which is within the comfort range.

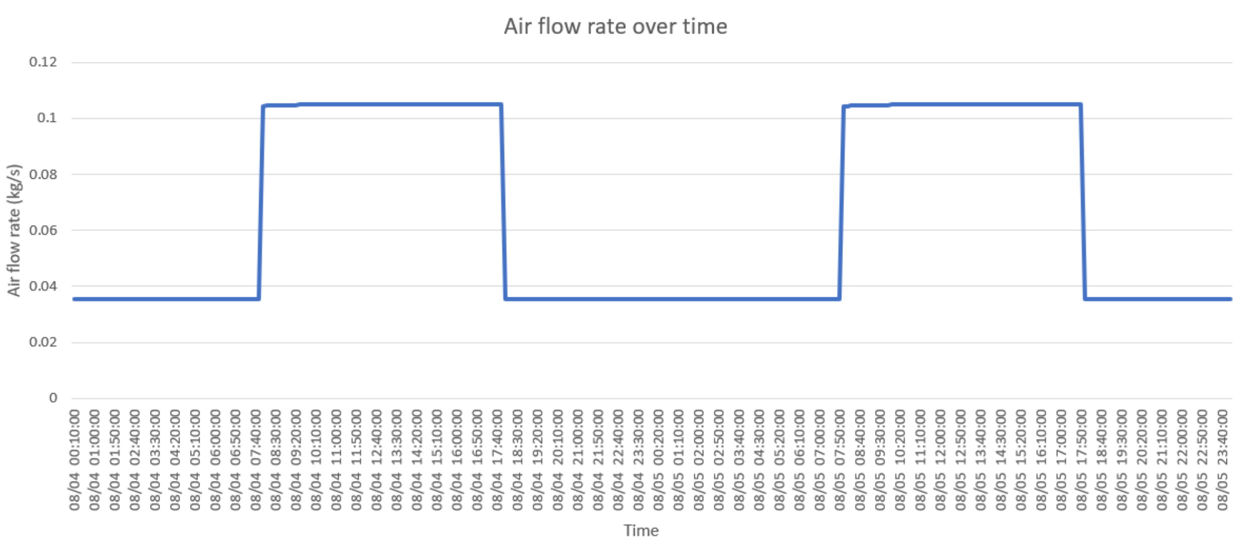


Figure 7.3.4 Air flow rate over time for voting-based HVAC control logic

7.4 Comparison between baseline and proposition

Compare the PMV between baseline and proposition, as shown in figure 7.4.1, we can notice that during the occupied period, even both baseline and proposition have 100% thermal comfort, however, PMV for proposition is more reasonable. For proposition, PMV largely falls between -0.1 and 0.1, and its average is -0.05678 (very close to zero). Under such thermal condition, occupants will not feel a bit chiller anymore and their satisfaction will be better.

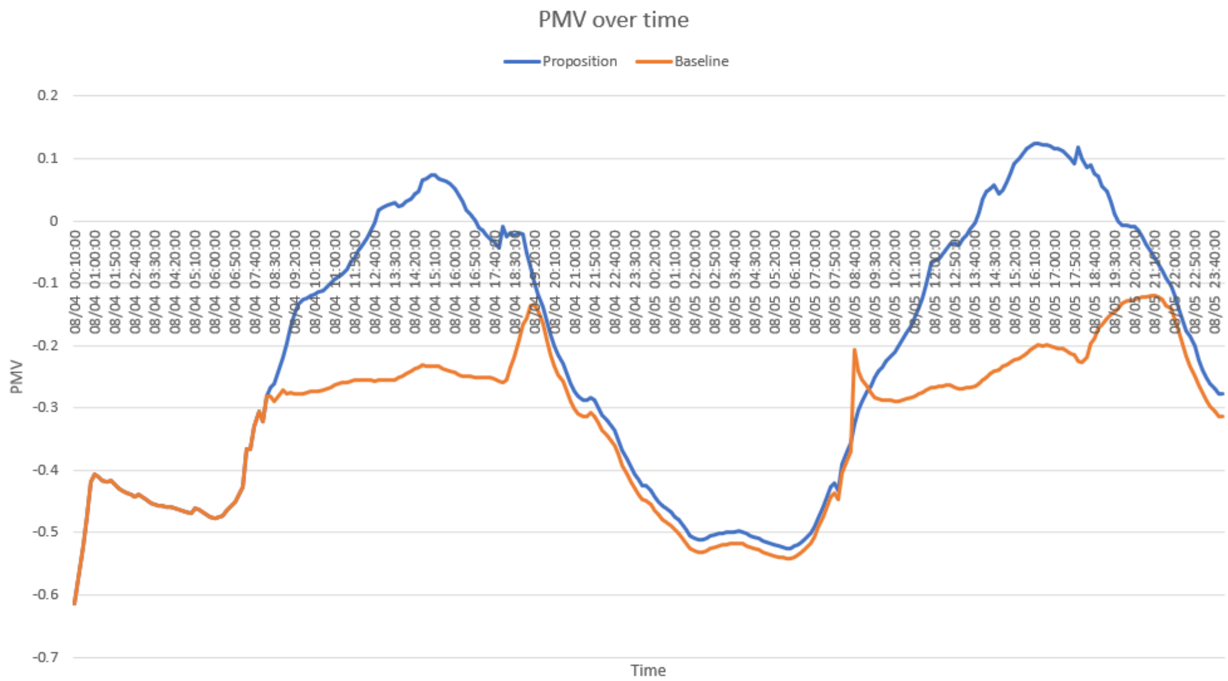


Figure 7.4.1 PMV comparison between baseline and proposition

In conclusion, the proposition solves the problem (occupants feel a bit chiller) identified during post occupancy evaluation (POE), therefore, it a successful proposition.

8 Limitations and Future Work

The limitations associated with this study are listed in this section, along with aspects which could be addressed in future work to take this project further.

Due to limitations of time and knowledge, the system could not be deployed on a server. As discussed earlier, an ideal implementation would have been to collect information or feedback from individual workstations and collect it at a central server where the control logic is housed. However, only a demonstration was explored, where all five interfaces were created on the same workstation, which served as a local "server" location. In the future, the system could be implemented on a server, enabling its verification via field experiments rather than a single zone simulation.

Although the same logic is proposed to be applied to the entire building in regularly occupied zones, only one of the zones was modeled using the proposed logic and verified via the simulation on EnergyPlus. Due to this, the energy performance before and after the simulation could not be compared, since EnergyPlus only provides building level Energy Consumption.

The simulation period on EnergyPlus was chosen over the period of only one month (August) in the cooling season. Ideally, the simulation would be verified in heating, cooling as well as swing seasons. This further verification and fine tuning can be done in the future.

The EnergyPlus model used required TMY data, which is not very accurate or representative of the microclimate around the building. This is a limitation of this project, which would be more accurate if local weather data could be used. This can be done in the future, to fine-tune the control logic while analyzing outputs for different seasons.

Another aspect of this project which can be done in the future is an improvement in the recommendation logic. Currently, the logic is fairly primitive, relying on binary values (hot or cold) to give recommendations which do not vary in intensity. This logic could be fine-tuned to make the recommendations more relevant or appropriate.

9 Bonus - User interface web application

There are total 5 user interface apps for the 5 occupants in the zone. All of them have been complied in MATLAB Web App Compiler app. Though installing and configuring MATLAB Web App Server on a local laptop, those apps can be accessed from a web browser using a URL. The communication protocol is HTTP/HTTPS. For each occupant, the corresponding App URL is transferred to a particular quick response (QR) code, so it is convenient for occupants to get into their vote app through scanning QR code. For facility manager, the URL is a home page listing all hosted user interface web apps, as well as facility interface web app. Therefore, the manager can not only control cooling setpoint in facility interface app, but also, access to user interface to diagnose control logic problems (which is actually what we did).

Table 7.4.1 Web apps for facility manager and each occupant

User	URL	QR code
Facility Manager	http://ruijis-mbp.wv.cc.cmu.edu:9988/webapps/home/	
Occupant 1	http://ruijis-mbp.wv.cc.cmu.edu:9988/webapps/home/session.html?app=Occupant1_App	
Occupant 2	http://ruijis-mbp.wv.cc.cmu.edu:9988/webapps/home/session.html?app=Occupant2_App	
Occupant 3	http://ruijis-mbp.wv.cc.cmu.edu:9988/webapps/home/session.html?app=Occupant3_App	
Occupant 4	http://ruijis-mbp.wv.cc.cmu.edu:9988/webapps/home/session.html?app=Occupant4_App	
Occupant 5	http://ruijis-mbp.wv.cc.cmu.edu:9988/webapps/home/session.html?app=Occupant5_App	

10 Conclusions

This study was a continuation of previous studies on the Center for Sustainable Landscapes (CSL) building. Based on prior Post Occupancy Evaluation results, a new temperature control logic was proposed. The aim of the proposal was to integrate occupants as active participators in the building's thermal comfort conditions. To do so, a user interface was designed which could obtain feedback from the users regarding their thermal sensation. The feedback was collected using a 7 point scale, ranging from -3 to +3. Each number on the scale was related to a linguistic temperature sensation label, ranging from "too cold" for -3 to "too hot" for +3. The users were shown only the linguistic values. The users were also provided some recommendations regarding the use of personal comfort devices, which they were free to accept or reject. The user votes were aggregated every ten minutes. Based on the results of this aggregation, the zone temperature setpoint was updated.

The user interfaces for both the occupants and the facility managers were created using the Matlab App Designer, and the control logic was implemented using Simulink. Using the co-simulation platform MLE+, the Simulink logic was implemented on an EnergyPlus model. The simulation was run for the month of August, in the cooling season. Predicted mean vote (PMV) values were used as a metric to compare the baseline simulation which was run without the control logic, and the simulation which used the control logic. From the results of the analysis, it was found that the PMV values for the proposed control logic were closer to zero, indicating improved thermal comfort for occupants. Hence, the results of our simulation were positive, and the control logic was found to work.

11 References

[1] U. E. I. Administration, "Preliminary end use consumption estimates," Commercial Building End Use Consumption Survey, 1999, 2003

[2] Henneberger, John, and Michael Scelzi. "Temperature control voting system." U.S. Patent No. 9,989,944. 5 Jun. 2018.

[3] Erickson, Varick L., and Alberto E. Cerpa. "Thermovote: participatory sensing for efficient building hvac conditioning." Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings. ACM, 2012.

[4] Jazizadeh, Farrokh, et al. "Human-building interaction framework for personalized thermal comfort-driven systems in office buildings." Journal of Computing in Civil Engineering28.1 (2013): 2-16.

[5] https://www.cs.princeton.edu/courses/archive/fall13/cos521/lecnotes/lec8.pdf?fbclid=IwAR2xKOltDw1QnRmPpQeKDuyc8jSoaMwO7omplF6AbKZ1Xvp_PtqNU-5Sjhc

12 Appendix - Worklog

All team members discussed and identified the control problem as a group.

Ruiji Sun:

- Control Logic Design and Implementation in Simulink (major)
- EnergyPlus Configuration
- Web Application for User Interface

Suzy Li:

- User Interface Design
- Control Logic Design

Rashmi:

- Literature Review
- Control Logic Design and Data Communication
- Report Compilation

Tiancheng:

- Data Communication
- Control Logic Implementation in Simulink
- Simulation Results Analysis