



**NIST Technical Note
NIST TN 2298**

Development of a Simulation Tool for the Intelligent Building Agents Project

Dr. Amanda J Pertzborn

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.TN.2298>

**NIST Technical Note
NIST TN 2298**

Development of a Simulation Tool for the Intelligent Building Agents Project

Amanda J. Pertzborn, PhD
*Building Energy and Environment Division
Engineering Laboratory*

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.TN.2298>

August 2024



U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

NIST TN 2298
August 2024

Certain commercial equipment, instruments, software, or materials, commercial or non-commercial, are identified in this paper in order to specify the experimental procedure adequately. Such identification does not imply recommendation or endorsement of any product or service by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

NIST Technical Series Policies

[Copyright, Use, and Licensing Statements](#)

[NIST Technical Series Publication Identifier Syntax](#)

Publication History

Approved by the NIST Editorial Review Board on 2024-07-31

How to Cite this NIST Technical Series Publication

Pertzborn, Amanda. (2024) Development of a Simulation Tool for the Intelligent Building Agents Project. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Technical Note (TN) 2298.
<https://doi.org/10.6028/NIST.TN.2298>

NIST Author ORCID iDs

Amanda Pertzborn : 0000-0002-1473-7500

Contact Information

amanda.pertzborn@nist.gov

Abstract

In the U.S., commercial buildings are responsible for approximately 36 % of total energy consumption, and the heating, ventilation, and air-conditioning (HVAC) systems make up about 52 % of that total. Improving building operations can significantly reduce the amount and cost of the energy used in the commercial building sector. The Intelligent Building Agents Laboratory (IBAL) at the National Institute of Standards and Technology (NIST) was designed to emulate the air and hydronic systems in a small commercial building primarily to develop and study advanced control approaches for HVAC systems. This report focuses on the calibration and validation of a simulation tool that couples a model of the IBAL with a virtual building model. That tool, IBASIM, will be used to quickly evaluate different control approaches before selecting and implementing the most promising approaches in the IBAL.

Keywords

Calibration; controls; co-simulation; HVAC; intelligent building agents; simulation; validation.

Table of Contents

1. Introduction	1
2. Model Overview	3
2.1. IBAL Data	5
2.2. HSIM	12
2.3. ISIM	12
2.4. Model Limitations	16
3. Component Calibration and Validation	18
3.1. Metrics	18
3.2. Component level calibration	19
3.2.1. Chiller models	19
3.2.2. Hydronic flow, pressure, and pump power	27
3.2.2.1. Overview of the flow calculations	27
3.2.2.2. Parameters for the flow calculations	30
3.2.2.3. Pump power consumption	31
3.2.2.4. Validation	32
3.2.3. AHU fan power consumption	34
3.2.4. AHU cooling coil	35
4. Added Controllers	38
4.1. Valve controllers	38
4.2. Secondary loop operation	39
4.3. Chiller staging	40
4.4. TES operation	43
4.5. Chilled water temperature setpoint	44
4.6. Airflow setpoint	45
4.7. AHU static pressure setpoint	45
4.8. Candidates for advanced control approaches	46
5. System-level Validation	47
5.1. Overall performance	48
5.2. Component performance	53
5.2.1. Chiller models	54
5.2.2. Hydronic flow	55
5.2.3. Pump power	57
5.2.4. AHU fan power	57

5.2.5. AHU cooling coil	58
5.2.6. Control decisions	59
6. Conclusion	61
References.....	62
Appendix A. IBAL Data Dictionary	64
Appendix B. List of Symbols, Abbreviations, and Acronyms.....	67

List of Tables

Table 1. Setpoints from HSIM.....	12
Table 2. IBAL controllers and measurements implemented in the MATLAB portion of ISIM.....	16
Table 3. Variable definitions for chiller equations	20
Table 4. Lower and upper bounds on chiller parameters for optimization.	21
Table 5. Calibrated parameter values for chiller models.....	23
Table 6. Datasets for chiller calibration.	23
Table 7. Parameter values for the valves in <i>Type 9031</i>	31
Table 8. Parameter values for the pump differential pressure in <i>Type 9031</i>	31
Table 9. Parameter values for the pump power in <i>Type 9031</i>	32
Table 10. Datasets for validating the AHU fan power models.	34
Table 11. Parameters in the cooling coil models for the AHUs.	35
Table 12. Settings for the PI logic for valves.	38
Table 13. Settings for the PI logic for the VAVs in cooling mode.	45
Table 14. Test cases for system-level validation. All test locations are Atlanta, GA, US.....	47
Table 15. Number of outliers as a percentage of the total data points and the median percent error.	49
Table 16. Models for further development.....	61
Table 17. Definitions of measurements.	64

List of Figures

Figure 1 Components of the entire IBASIM simulation space.....	5
Figure 2 Schematic of the hydronic system, with a focus on the primary loop.	7
Figure 3 Schematic of the hydronic system, with a focus on the secondary loop.	8
Figure 4 Overview of the air system.	9
Figure 5 Focus on the AHUs in the air system.	10
Figure 6 Focus on the VAVs in the air system.	11
Figure 7 Overview of the TRNSYS model implemented in Simulation Studio.	14

Figure 8 Details of the (a) AHU1 and (b) AHU2 macros.	15
Figure 9 Chiller model results.	24
Figure 10 Comparison between data and simulation model for chiller power and chilled water temperature.	25
Figure 11 Comparison between data and simulation model for chiller power and chilled water temperature with large transients removed.	25
Figure 12 Close-up of a period of cycling in Chiller1.	26
Figure 13 Pseudocode describing the process to calculate the flow rates in the SL.	28
Figure 14 Pseudocode describing the process to calculate the flow rates in the SL.	28
Figure 15 Pseudocode for calculating flows in the PL.	29
Figure 16 Pump power results.	33
Figure 17 Comparison of the simulation results for the pump powers to the measured values.	33
Figure 18 Results for AHU fan power consumption model.	35
Figure 19 Results for AHU cooling coil models.	37
Figure 20 Pseudocode for setting the bias term for the cooling coil valves.	39
Figure 21 Pseudocode for determining if there is a call for cooling that requires the secondary loop to operate.	40
Figure 22 Pseudocode for chiller staging.	42
Figure 23 Pseudocode for determining if the chiller stage should be changed.	43
Figure 24 Pseudocode for determining the chilled water temperature setpoint, which is used by the chiller and the SL.	44
Figure 25 Boxplots of the percent error in the power between the simulation and the data. The y-axes cut off some of the outliers.	49
Figure 26 Normalized total power consumption.	50
Figure 27 Normalized power for individual components.	51
Figure 28 Normalized cost.	52
Figure 29 Airflow and temperature data related to the underprediction of fan power during the pre-cooling period for the <i>ex_shift</i> case.	53
Figure 30 Chiller1 performance for seven cases under system-level validation.	54
Figure 31 Chiller2 performance for seven cases under system-level validation.	55
Figure 32 Hydronic flow results when used in the system-level validation.	56
Figure 33 Valve model results when used in system-level validation.	56
Figure 34 Pump power results when used in system-level validation.	57
Figure 35 AHU fan power results when used in system-level validation.	57
Figure 36 Performance metrics for the inputs to the AHU1 fan power model.	58
Figure 37 Performance metrics for the inputs to the AHU2 fan power model.	58

Figure 38 AHU cooling coil results when used in system-level validation. 59
Figure 39 Performance of the on/off control decisions. 60
Figure 40 VAV airflow rate results when used in system-level validation. 60

Acknowledgments

The work in this report was partly funded by NIST grants 70NANB17H277, 70NANB18H203, and 70NANB21H108.

1. Introduction

In the U.S., residential and commercial buildings are responsible for approximately 36 % of total energy consumption, and the heating, ventilation, and air-conditioning (HVAC) systems make up about 52 % of that total [1]. Improving building operations can significantly reduce the amount and cost of the energy used in the commercial building sector. The Intelligent Building Agents Laboratory (IBAL) at the National Institute of Standards and Technology (NIST) was designed to emulate the air and hydronic systems in a small commercial building primarily to develop and study advanced control approaches for HVAC systems. However, the IBAL also has other research uses, including as a source of data for the creation of machine learning (ML) models of standard HVAC equipment [2, 3], a test case for the development of semantic models of HVAC systems [4], and a well-characterized and controlled system for the development and testing of ML-based fault detection and diagnostics (FDD) algorithms.

This report focuses on the calibration and validation of a simulation tool that couples a model of the IBAL with a virtual building model. That tool, IBASIM, will be used to quickly evaluate different control approaches before selecting and implementing the most promising approaches in the IBAL. This tool can be obtained online from Ref. [5]. The IBAL contains standard HVAC equipment that can be categorized as part of the air or hydronic systems. The air system includes two air handling units (AHUs), four variable air volume terminal units (VAVs), an outdoor air unit (OAU), and four zone simulators or emulators (ZSs). The hydronic system includes two chillers and ice-on-coil thermal energy storage (TES). The two systems are coupled by the cooling coils in the AHU. The working fluid in the hydronic system is a 30 % propylene glycol (PG) mixture. The loads generated in the air system are met by either the chillers or the TES. The TES operates in three modes. In charge mode, a chiller produces PG at a temperature below the freezing point of water so that as it passes through the TES, ice builds. In discharge mode, PG from the cooling coils passes through the TES, melting the ice and providing cold PG to the cooling coils to meet the emulated loads. In bypass mode the TES is not used.

The OAU generates repeatable weather conditions for the incoming building air, regardless of the actual weather conditions. The ZSs generate repeatable building loads with electric heaters for sensible loads and steam spray humidifiers for latent loads. The OAU and ZS allow for the generation of repeatable boundary conditions, so different control approaches can be compared under the same weather and load conditions. Details about the design of the IBAL can be found in Refs. [6–8].

This report discusses the development of a simulation model of the IBAL coupled with a virtual building model. The IBAL operates in real-time, i.e., one minute is one minute, whereas a simulation can run much faster than real-time, as quickly as the calculations can be performed. The overall goal is to have a platform where new approaches for control can be tested and modified quickly. Then, the most promising approaches can be implemented in the IBAL using

actual equipment. This process takes advantage of the time efficiency of a simulation. However, since every simulation makes simplifying assumptions about the operation of actual equipment, the ultimate assessment of the control approaches must be made using actual equipment. The IBASIM tool does not explicitly model the OAU and ZSs. The key components of the hydronic system that are modeled include the two chillers and the TES. The key components modeled in the air system include the two AHUs and four VAVs.

The simulation was developed through grants to universities [9–12] and additional work at NIST. To ensure that the simulation adequately captures the behavior of the IBAL, the parameter values used to characterize the component models were calibrated to match measured data, and then the models were individually validated against data. The components were combined into the full simulation platform, and a system-level validation was performed. Although the result is a validated simulation tool, it has limitations. Some component models do not capture the full range of operations or the dynamics of the actual equipment. Although these models are sufficient for initial development work, they will be re-calibrated and re-validated as necessary. New or modified models will also be developed if the present models are inadequate for the control approaches being studied.

This report begins in Sec. 2 with an overview of the data used to calibrate and validate the component models. In addition, it presents the overall simulation architecture and discusses the components that make up the simulation. Section 3 details the component-level calibration, validation, and metrics used to evaluate the models. Section 4 discusses the controllers implemented in the simulation, and Sec. 5 presents the system-level validation. Conclusions and future work are discussed in Sec. 6. The Appendices include a data dictionary and list of nomenclature.

2. Model Overview

This section provides details about the model and data. As already mentioned, a simulation of the IBAL was built over several years through a series of grants to different universities. To make that model fully functional, some modifications were required, including coupling that model with a building model that provides realistic building loads based on simulated weather and occupancy and the response of the building conditions to the operation of the air and hydronic systems. This approach follows one developed as part of the “Hardware-in-the-loop Laboratory Performance Verification of Flexible Building Equipment in a Typical Commercial Building” research project [13]. That project is referred to as HILFT (hardware-in-the-loop load flexibility testing) throughout the rest of this report. The HILFT project generated operational data for an emulated commercial building under different control strategies and climate conditions to better understand how to use buildings to support grid needs [14]. The basic load flexibility strategies examined in HILFT are:

- eff – efficiency baseline case using ASHRAE Standard 90.1-2004 [15],
- shed – load shedding, which uses zone temperature setpoint changes to decrease the zone loads during the peak period for electric rates,
- shift – load shifting, which pre-cools the zones so that setpoints can increase during the peak period for electric rates without compromising occupant comfort,
- shift with TES – charge the TES during the night and discharge the TES during the peak period for electric rates,
- modulating – operate the AHU fans for frequency regulation, and
- mpc – model predictive control instead of the rules-based control used in the other cases.

These strategies were tested using a virtual building model (VBM) set in four locations in the US – Atlanta, GA; Buffalo, NY; New York City, NY; and Tucson, AZ – coupled with the IBAL. Three different weather conditions were selected for each location: a typical shoulder day, a typical summer day, and an extreme summer day. Over 100 test cases were completed in the IBAL.

A vital part of HILFT was the development of the VBM, an occupancy behavior model (OBM), and necessary supervisory level controllers (Sec. 4). These elements are referred to as HSIM (HILFT Simulation). The VBM uses the Medium Office DOE Prototype Building [16] implemented in EnergyPlus [17] and is discussed in more detail in Sec. 2.2. Since the IBAL has four zones, the HILFT project selected four of the zones in the building model based on the maximum load that each zone emulator in the IBAL can generate. The EnergyPlus building model uses an ideal load calculation to determine the temperature and humidity of the zone given the internal and external (i.e., solar gain) loads, the thermal mass of the zone, the condition of the air entering the zone (temperature, humidity, and airflow rate), and the zone setpoints. The basic

procedure is to pass the actual air temperature, humidity, and airflow rate from the IBAL to the VBM, which then calculates the zone temperature and humidity, and these values are used as the setpoints for the next minute for the zone emulators in the IBAL. Note that these setpoints are not the thermostat setpoints; they are setpoints used by the zone emulators to generate the zone loads from the VBM. This communication occurs every minute. In this way, the VBM provides the IBAL with a realistic thermal response of the zone to the condition of the air generated by the HVAC system, and the IBAL gives the VBM the real dynamics of the HVAC system.

In this work, the goal is to replace the real IBAL with a simulation called ISIM that is calibrated and validated using data from the IBAL. The IBAL model is developed in TRNSYS¹ (TRaNsient SYstem Simulation) [18], some IBAL controllers are implemented in MATLAB [19], the VBM uses EnergyPlus, the OBM uses MATLAB, and Simulink [20] is the co-simulation platform. The entire simulation package is called IBASIM and is publicly available [5].

Figure 1 shows IBASIM and how the various components interact. The TRNSYS simulation calls a MATLAB script, which launches the other simulation pieces. The elements that are part of ISIM and HSIM are labeled as such. The TRNSYS simulation is called TSIM. Some elements contain pieces that are linked to both development efforts. For example, some controllers were developed as part of the HILFT project, but others were created as part of the ISIM effort. IBASIM's simulation timestep is one minute. In summary, ISIM is a stand-in for the actual laboratory, and HSIM is a stand-in for an actual building. ISIM and HSIM contain elements that act as stand-ins for actual controllers; the placement is based on where it makes the most sense to place that controller for simulation purposes. The following sections include details about the IBAL and how the data were generated, as well as a high-level discussion of the HSIM and ISIM components.

¹ Certain equipment, instruments, software, or materials, commercial or non-commercial, are identified in this paper in order to specify the experimental procedure adequately. Such identification does not imply recommendation or endorsement of any product or service by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

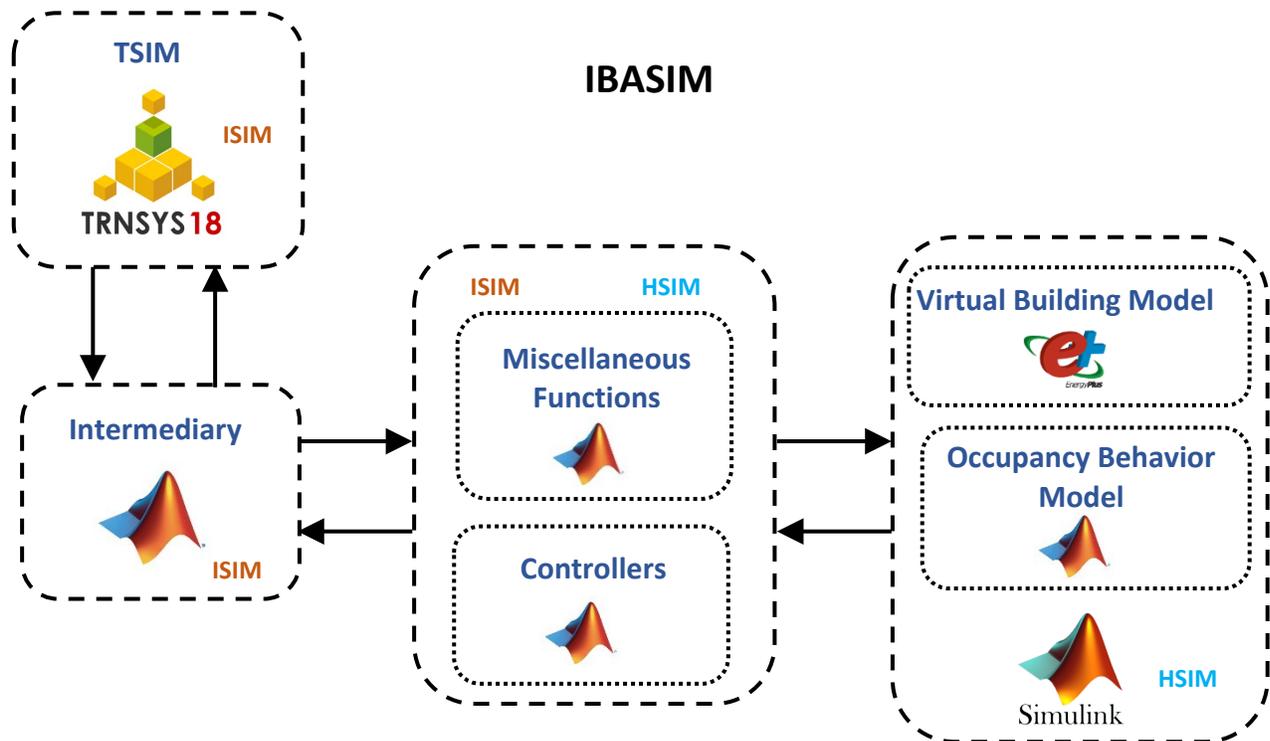


Figure 1 Components of the IBASIM simulation platform.

2.1. IBAL Data

Data are collected in the IBAL at a 10-s sampling interval (0.1 Hz). Two types of datasets were generated to calibrate and validate the models presented in this report: component data and system data. The component data are specific to an individual piece of equipment. For example, some data used to calibrate and validate the TES model were developed via tests focused on the TES's charge and discharge operating modes. Estimates of the pressure drop through valves and dampers were achieved by carrying out tests that operated those components under a range of conditions. System-level data were generated considering multiple pieces of equipment and are generally those developed as part of the HILFT project.

All data discussed in this report can be obtained from the [IBAL Database](#) [21]. The database contains two dashboards, "Experiments" and "Measurements." The Experiments dashboard allows the user to search for a specific experiment. For example, the title of all experiments related to the HILFT project is "DOE Formal Testing," so searching by that experiment title will present those tests. Each test is named based on the date when it started. For example, 2022_05_05_run2 is the HILFT test corresponding to an extreme summer day in Atlanta, GA, using an operational approach and building design that is compliant with ASHRAE Standard 90.1-2004 that was carried out on May 5, 2022, and it was the second test that day.

Figures 2 and 3 show the major components in the hydronic loop and the locations of key measurements. Figures 4 through 6 show the major components of the air system and key measurements. The measurements are defined in Appendix A and include measurements of pressure (p), differential pressure (dp), power ($power$), temperature (rtd), relative humidity (rh), and flow (f).

The hydronic system is a primary-secondary design with a hydraulic bridge that decouples the two loops and controls the temperature of the secondary loop (SL). The mixing valve, V9, actuates so that the right amount of PG from the outlet of the secondary loop mixes with PG from the outlet of the primary loop (PL) to produce the setpoint temperature at the inlet of the secondary loop as measured by *pump3_out_rtd*. The pipe on the right of the bridge contains a check valve that only allows flow through that pipe if the SL flow is greater than the PL flow. In that scenario, the excess PG in the SL returns to the inlet of the SL and mixes with the PG from the PL. Figure 2 focuses on the components in the PL, and Fig. 3 highlights the components in the SL. The components labeled V# are valves. One of the components in the SL is HX1, which is a heat exchanger with one side connected to hot water and the other side connected to the PG flow in the SL. HX1 acts as an additional cooling load, essentially emulating the cooling coil in an AHU. If the hot water valve is closed, then HX1 provides a flow path so that Pump3 is not deadheaded if the cooling coil valves are closed but the pump is on.

Figure 4 shows that each AHU serves two VAVs, with each VAV serving one zone (ZS). The components labeled D# are dampers. The emulated outdoor air (EOA) is not modeled; it is represented by a temperature and humidity for every minute of the day provided by the VBM (Sec. 2.2) and generated by the OAU. The VBM also provides load setpoints for the ZSs. A portion of the air leaving the ZSs recirculates to the inlet of the AHUs, where it mixes with the EOA. Figure 5 shows the AHUs in more detail, with the key measurements noted. Each AHU has an electric preheater to ensure that the inlet air will not freeze the fluid in the cooling coil, but this heater is not currently used in the IBAL. Figure 6 shows the four VAVs and their key measurements.

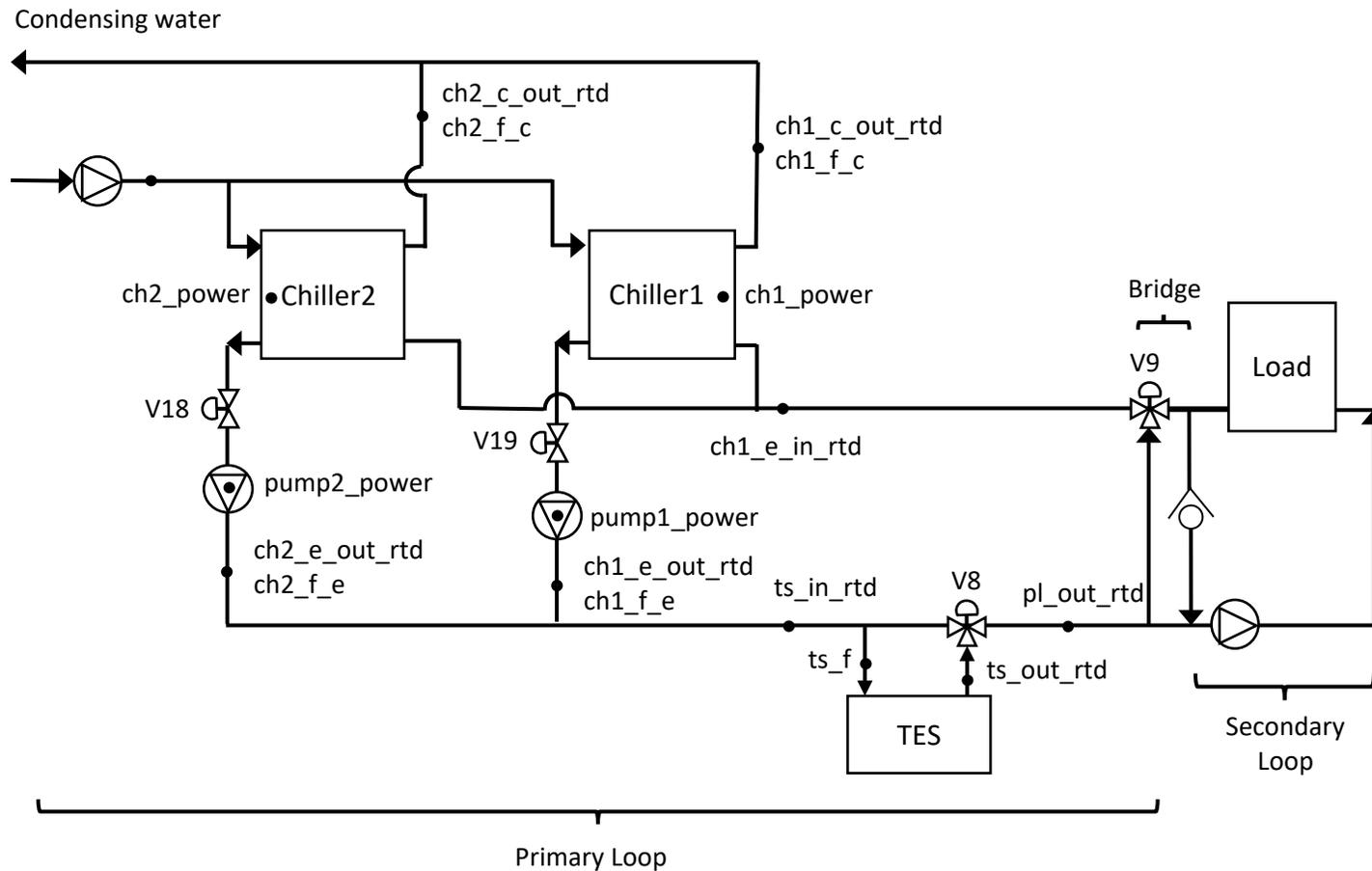


Figure 2 Schematic of the hydronic system, with a focus on the primary loop.

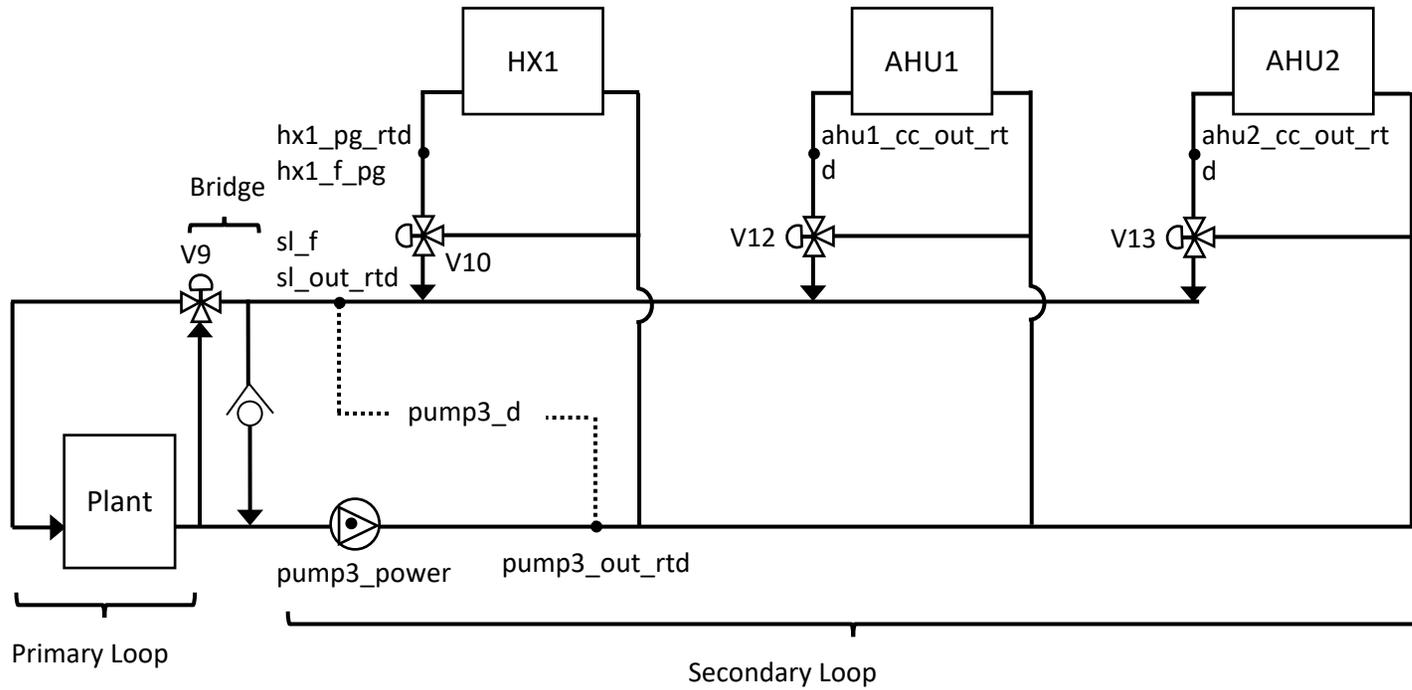


Figure 3 Schematic of the hydronic system, with a focus on the secondary loop.

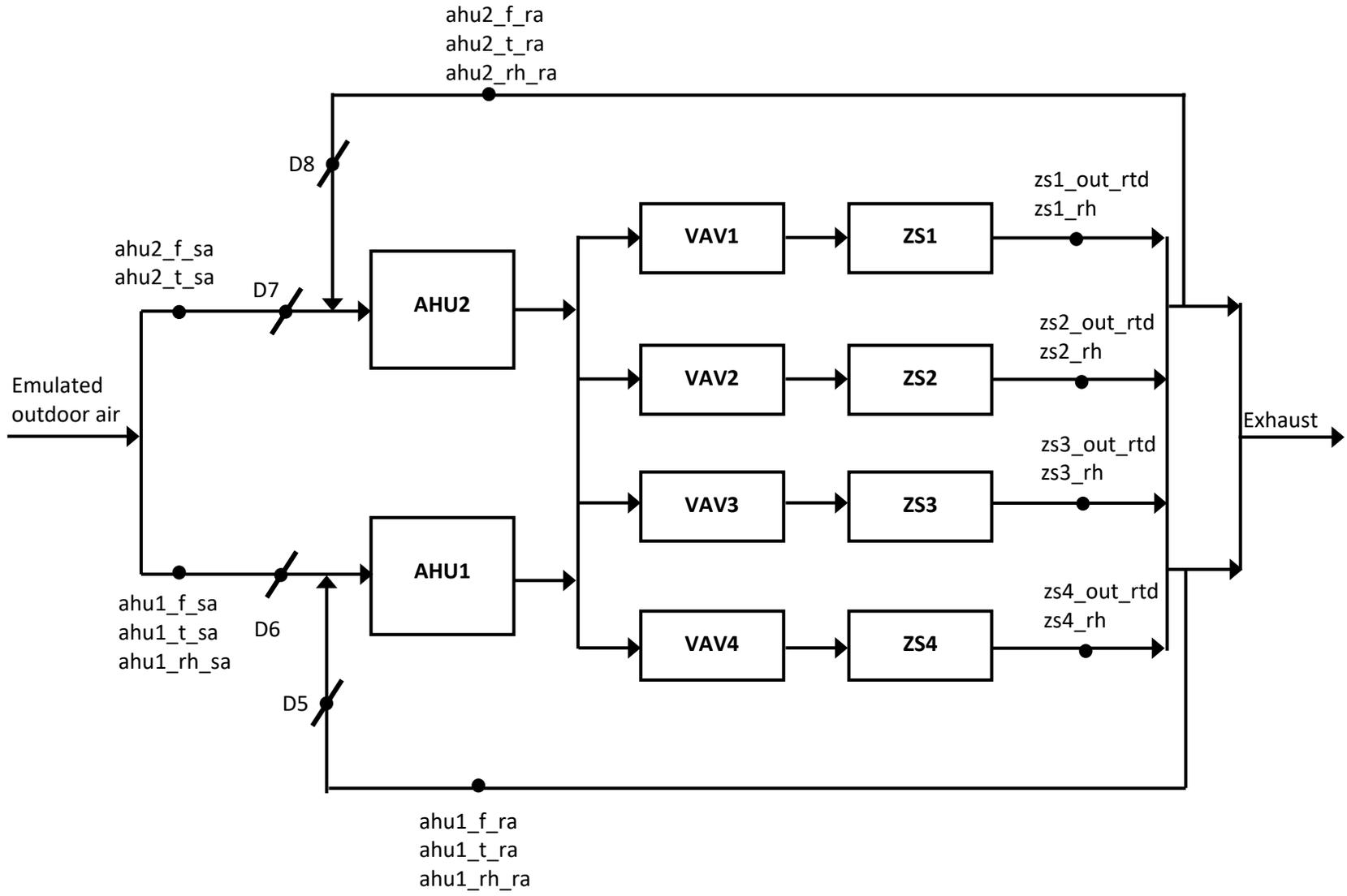


Figure 4 Overview of the air system.

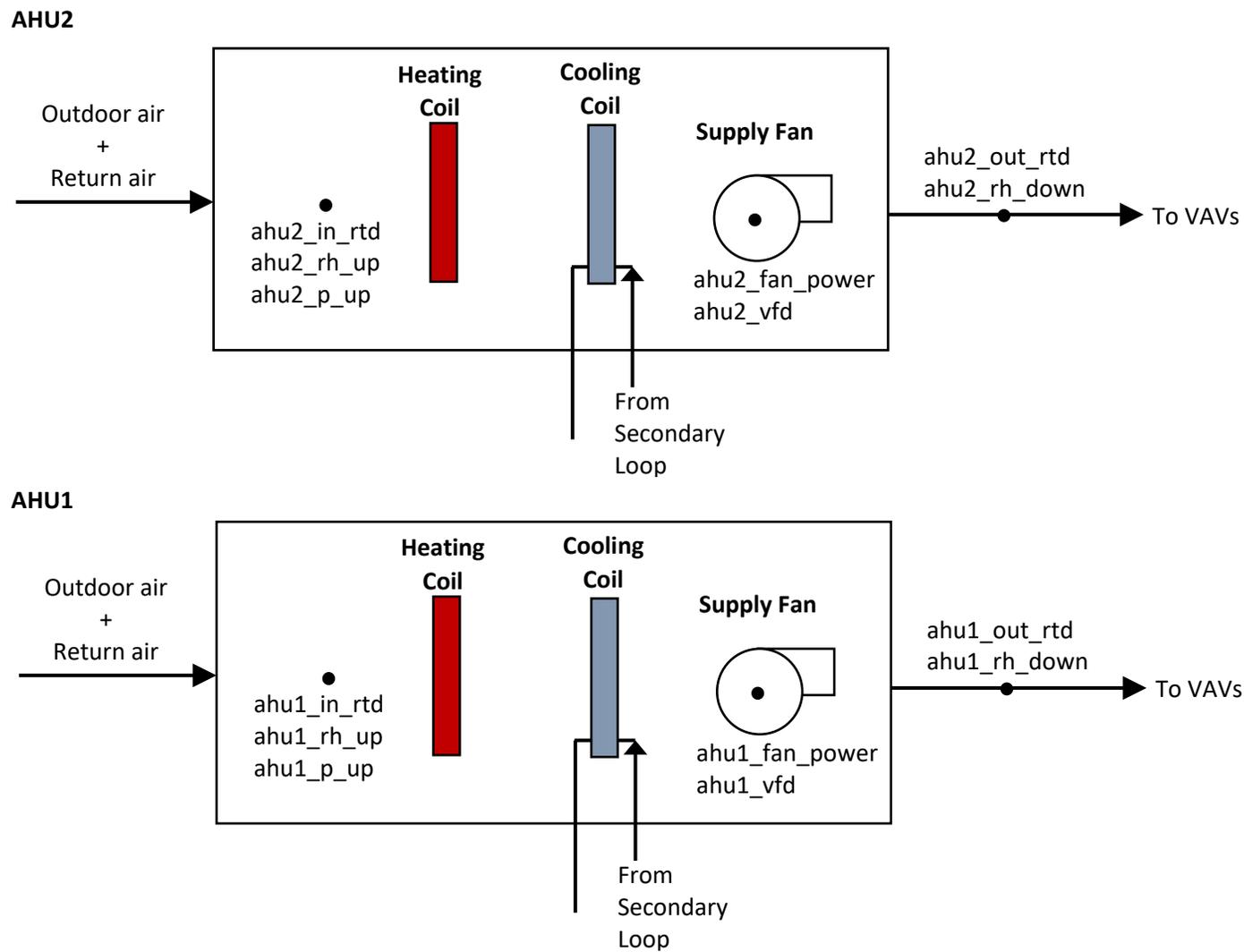


Figure 5 The AHUs in the air system.

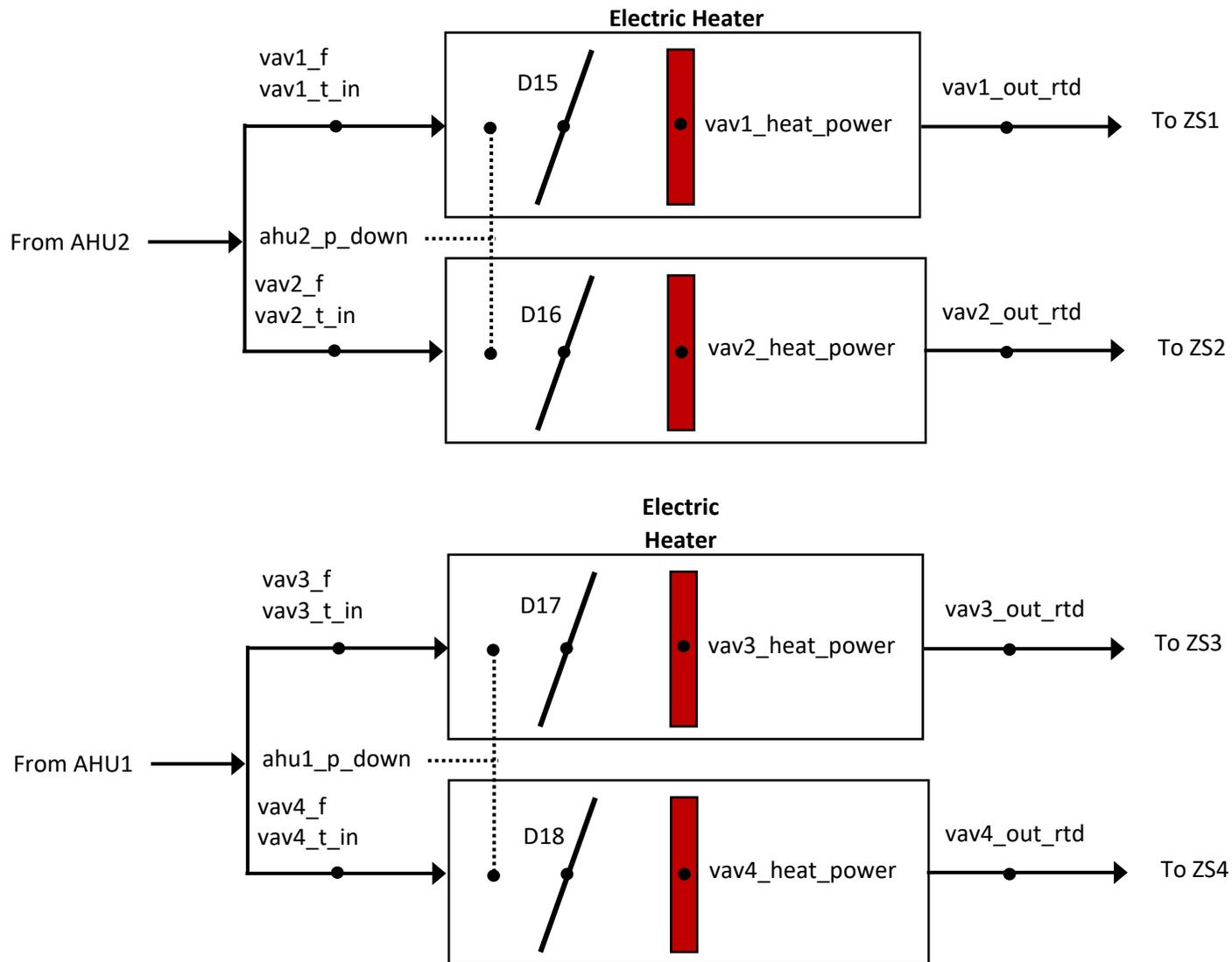


Figure 6 The VAVs in the air system.

2.2. HSIM

HSIM is adapted from the HILFT project. The VBM is a medium-sized office building [16] modeled in EnergyPlus and called from Simulink using a functional mockup unit (FMU) [22]. The four zones selected for the HILFT project are two enclosed offices, one open office, and a conference room. The OBM was adapted from [23, 24] and implemented as a MATLAB script. This model allows virtual occupants to take actions, including changing a thermostat or turning on a space heater, which impact the zone loads. Table 1 lists the control setpoints that HSIM defines. The *Function Name* column lists the scripts that calculate the setpoint. When multiple names are listed, there are different scripts depending on the mode of operation. For example, if the TES is in use, the chilled water setpoint (T_{chwst}) is defined differently from a case where it is not used. The function names that are italicized were replaced in ISIM.

Table 1. Setpoints from HSIM.

Variable Name	IBAL Name	Description	Function Name
T_chwst	ch1_t_sp ch2_t_sp	Chilled water temperature setpoint	DefaultSettingIBAL <i>RB_T_chwst_RESET_typical</i> <i>RB_CHW_RESET</i>
DP_slSP	sl_vfd_sp	Chilled water secondary loop pressure setpoint	DefaultSettingIBAL <i>RB_CHW_RESET</i>
P_sp_ahu1	ahu1_fan_sp_inh2o	AHU1 supply air static pressure setpoint	DefaultSettingIBAL <i>RB_SAP_RESET_ahu1</i>
P_sp_ahu2	ahu2_fan_sp_inh2o	AHU2 supply air static pressure setpoint	DefaultSettingIBAL <i>RB_SAP_RESET_ahu2</i>
T_SA_ahu1	ahu1_cc_sp_f	AHU1 supply air temperature setpoint	DefaultSettingIBAL <i>RB_SAT_RESET_ahu1</i>
T_SA_ahu2	ahu2_cc_sp_f	AHU2 supply air temperature setpoint	DefaultSettingIBAL <i>RB_SAT_RESET_ahu2</i>
TS_mode	ts_mode	TES operating mode	Control_Model
Vmin_vav1_ahu1 Vmin_vav2_ahu1	ahu1_sa_sp_low	Minimum ventilation airflow rate for each zone	Vmin_RESET_z3 Vmin_RESET_z4
Vmin_vav1_ahu2 Vmin_vav2_ahu2	ahu2_sa_sp_low	Minimum ventilation airflow rate for each zone	Vmin_RESET_z1 Vmin_RESET_z2

2.3. ISIM

ISIM is composed of a TRNSYS model, TSIM, which captures most of the IBAL's thermal and hydraulic properties, and several MATLAB functions, which capture local and supervisory control actions and estimate fan power. The original TRNSYS models were developed in grants awarded to Drexel University and the University of Wisconsin-Madison [9–12].

Figure 7 is an overview of TSIM. This figure illustrates how components are linked, but for a better understanding of the program, see Ref. [5]. In TRNSYS, each component is modeled in a type, and each instance of a type is a unit. *Type 11*, for example, is a “Tee Piece,” which mixes

two liquid streams and calculates the thermal properties of the output; *Unit 58* is an instance of *Type 11*. A unit's output is connected to another unit's input by a line. The yellow lines connect units that are printers or plotters for saving or displaying the simulation results; those units are not shown in the figure. AHU1 and AHU2 are implemented as macros, the details of which are shown in Fig. 8. TRNSYS interacts with MATLAB via *Type 155*. At the end of a simulation timestep, the outputs from the types in TRNSYS are passed to MATLAB, the MATLAB components from HSIM and ISIM execute, and their outputs are returned to TRNSYS.

A base TSIM file was one of the deliverables from grant 70NANB21H108, but this model was modified to work in the IBASIM configuration. Some of the controllers implemented in TSIM were also implemented in HSIM, so the TSIM versions were removed. In addition, some controllers had been only partially created in TSIM (e.g., chiller staging) or were implemented but did not adequately capture the actual IBAL operations (e.g., cooling coil valve operation) and were therefore replaced or modified. Some existing types were also altered and re-calibrated on a more extensive dataset. These modifications are discussed in more detail in Secs. 3 and 4. Table 2 lists the quantities calculated by the MATLAB code in ISIM.

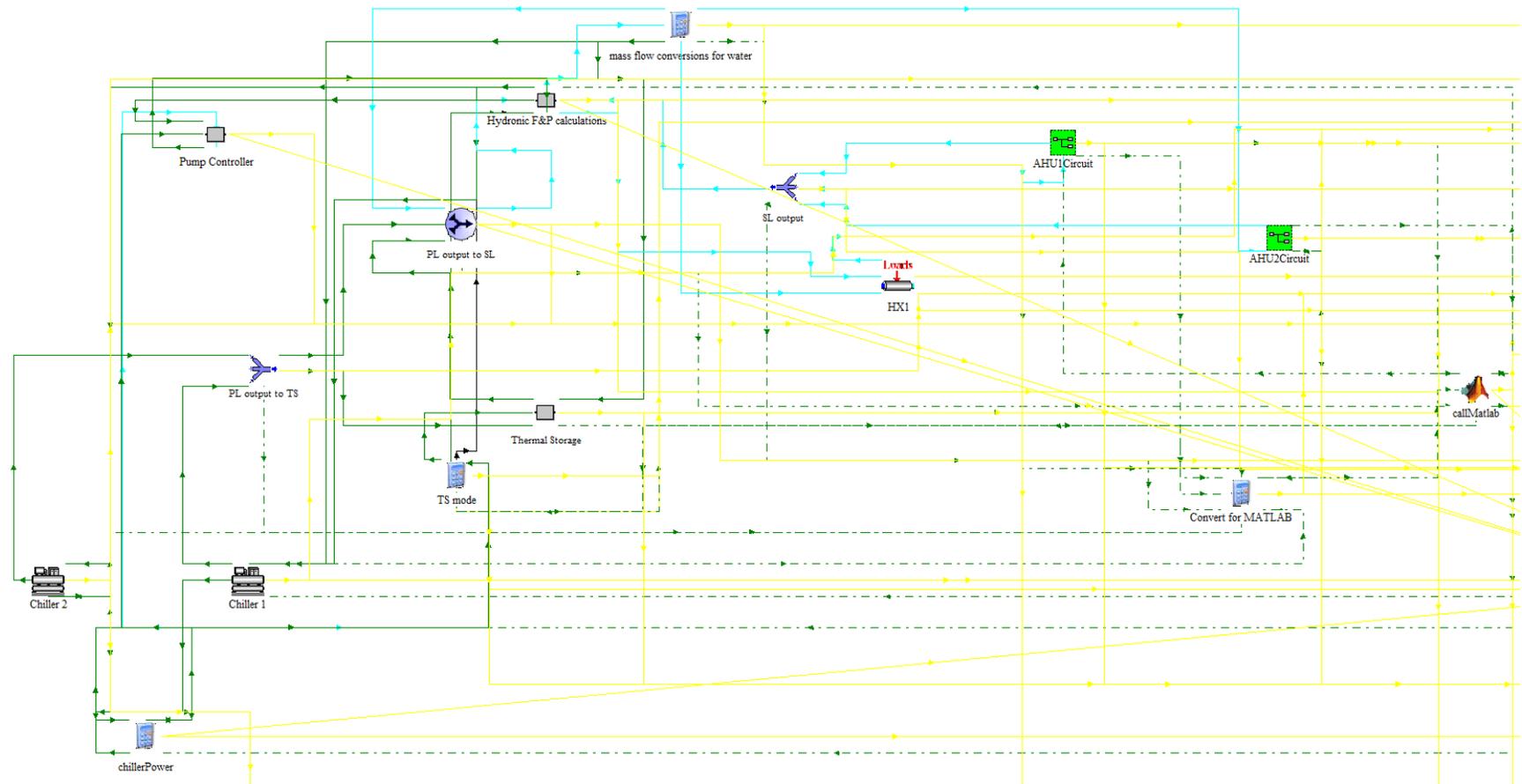
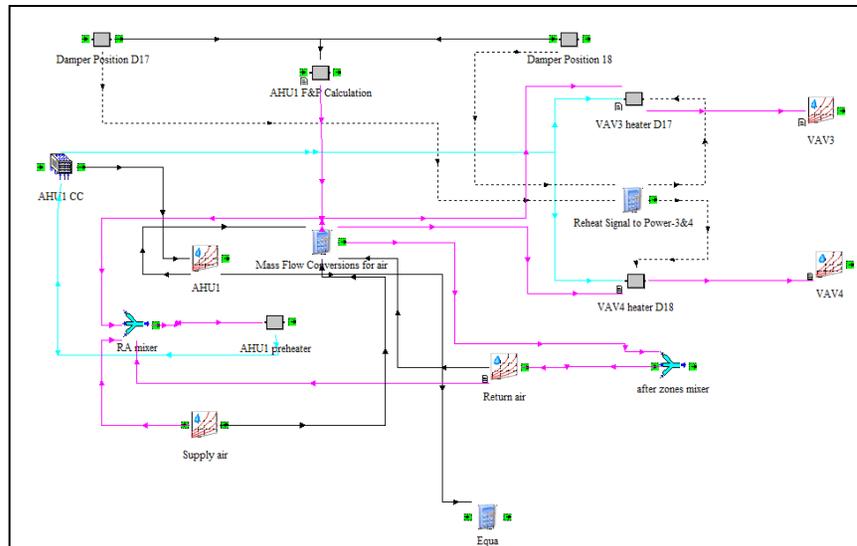
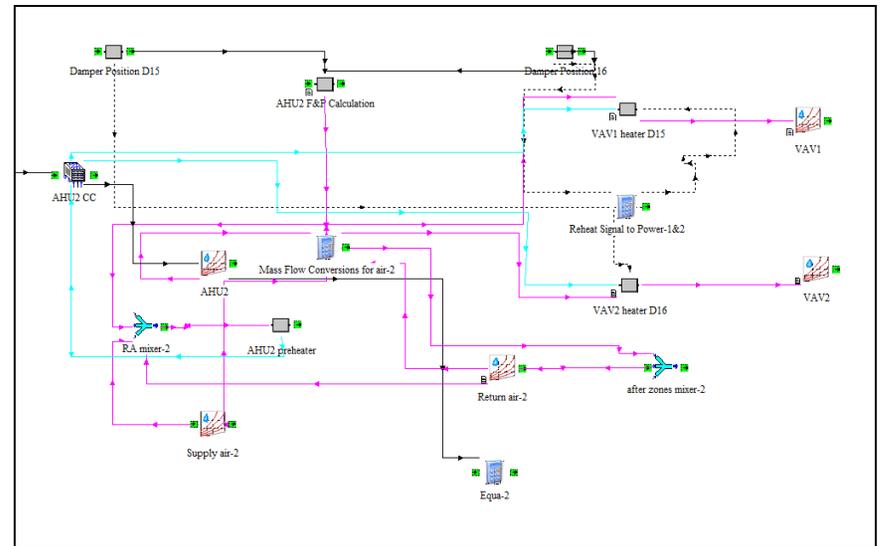


Figure 7 Overview of the TRNSYS model implemented in Simulation Studio.



(a)



(b)

Figure 8 Details of the (a) AHU1 and (b) AHU2 macros.

Table 2. IBAL controllers and measurements implemented in the MATLAB portion of ISIM.

Variable Name	IBAL Name	Description	Function Name
ch1On	ch1_on	If equal to 1, turns Chiller1 on	chillerStaging
ch2On	ch2_on	If equal to 1, turns Chiller2 on	chillerStaging
v8	v8_pos_c	Determines the position of the valve for the TES	tesValve
vav1_d_sp vav2_d_sp vav3_d_sp vav4_d_sp	vav1_f, vav2_f vav3_f vav4_f	Determines the airflow rate	vavControllers
Tz1SP Tz2SP Tz3SP Tz4SP	zs1_t_sp_f zs2_t_sp_f zs3_t_sp_f zs4_t_sp_f	Determines the zone temperature setpoint	vavControllers
ahu1VentAir ahu2VentAir	ahu1_sa_sp_low ahu2_sa_sp_low	Scales the minimum ventilation airflow rate from HSIM	callSim
ahu1_cc_valve ahu2_cc_valve	v12_pos_c v13_pos_c	Determines the position of the cooling coil valves	ahuCCValves
slOn	pump3_on	Determines if there is a call for cooling and, if there is, turns on the secondary loop	slController
P_sp_ahu1 P_sp_ahu2	ahu1_fan_sp_inh2o ahu2_fan_sp_inh2o	Calculates the pressure setpoint for the AHU fans	RB_SAP_RESET_ahu1_ajp RB_SAP_RESET_ahu2_ajp
ahu1_fan_power ahu2_fan_power	ahu1_fan_power ahu2_fan_power	Calculate fan power	ahu1FanPower_2023_07_06_v6 ahu2FanPower_2023_07_06_v6

2.4. Model Limitations

Several key differences exist between the behavior of the IBASIM model and real equipment. In general, the simulation responds much more quickly to changes than the actual equipment. In the IBAL, the zone sensible load is generated by modulating the control signal to an electric heater to generate the temperature specified by HSIM. This is the zone temperature. In the actual equipment, the measured value of the zone temperature is never precisely the temperature HSIM specifies, nor does it change instantly. However, in IBASIM, the zone temperature is immediately exactly that determined by HSIM.

In the IBAL, the airflow setpoint for each zone is calculated based on the difference between the measured zone temperature and the temperature setpoint corresponding to a thermostat setting. The air system modulates the VAV dampers and the AHU fan speed to get close to that

airflow setpoint, but they are never exactly equal. The calculation of the setpoint in IBASIM is essentially the same. However, there is an assumption that the airflow rate matches the setpoint and that it matches immediately. Real equipment has at least a slight lag between a change in the setpoint and the system's response. There are also some scenarios where the actual airflow rate cannot reach the setpoint due to equipment constraints.

The model also has difficulty matching the pressure drops in the system. This is partly because the IBAL does not have enough pressure transducers to measure the pressure drop across every branch of a valve or damper, every piece of equipment, or through every pipe or damper, so the models are calibrated based on a limited dataset. However, even if the IBAL had enough transducers, the measurement accuracy would be suspect in many locations because there is insufficient development length before and after obstructions. Some models use pressure to calculate flow or power, limiting their accuracy. The following sections discuss these issues in more detail.

3. Component Calibration and Validation

This section discusses the approach used to calibrate and validate the models at a component level. This overview will not include a detailed discussion of the work performed as part of the grants but instead focuses on the modifications and additions to the models made at NIST. When specific datasets from HILFT were used for calibration or validation, they are noted. For all other cases, the datasets were a combination of many smaller datasets and are not individually listed. All data are available from the [IBAL Database](#).

3.1. Metrics

The metrics selected for evaluating the accuracy of the simulations are based on ASHRAE Guideline 14 [25], referred to as G14. This guideline was developed to define an approach for evaluating the energy savings of building retrofits and includes directions for calibrating a simulation model, which could be used to predict energy use given specified retrofits. G14 uses the metrics defined in Eqs. (1) - (4) to evaluate the quality of model calibration. In these equations, y_{sim} is the output from the model, y_{data} is the measured result from the laboratory, \bar{y} is the mean value, and N is the number of samples.

- Mean bias error (MBE): calculates the mean offset between the model and the data. If MBE is less than zero, the model underpredicts the data on average.

$$MBE = \frac{\sum_{i=1}^N (y_{sim,i} - y_{data,i})}{N - 1} \quad (1)$$

- Normalized mean bias error (NMBE): normalizes MBE by the mean of the measured result; reported as a percentage.

$$NMBE = \frac{MBE}{\bar{y}_{data}} * 100 \% \quad (2)$$

- Root mean squared error (RMSE): calculates the mean error between the model and the data without regard to direction.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (y_{sim,i} - y_{data,i})^2}{N - 1}} \quad (3)$$

- Coefficient of variation root mean squared error (CVRMSE): normalizes the RMSE by the mean of the measured result. CVRMSE is reported as a percentage.

$$CVRMSE = \frac{RMSE}{|\bar{y}_{data}|} * 100 \% \quad (4)$$

NMBE and CVRMSE are not calculated for temperature because temperature uses a relative scale. For example, the CVRMSE for temperature in °C is very different from the value in K. As will be noted in the relevant sections, when NMBE and CVRMSE are presented for a temperature value, it is a temperature difference, as shown in Eqs. (5)–(8). In addition, RMSE is presented for temperatures to provide a deeper understanding of the errors.

$$MBE_{\Delta T} = \frac{\sum_{i=1}^N (\Delta T_{sim,i} - \Delta T_{data,i})}{N - 1} \quad (5)$$

$$NMBE_{\Delta T} = \frac{MBE_{\Delta T}}{\overline{\Delta T}_{data}} * 100 \% \quad (6)$$

$$RMSE_{\Delta T} = \sqrt{\frac{\sum_{i=1}^N (\Delta T_{sim,i} - \Delta T_{data,i})^2}{N - 1}} \quad (7)$$

$$CVRMSE_{\Delta T} = \frac{RMSE_{\Delta T}}{|\overline{\Delta T}_{data}|} * 100 \% \quad (8)$$

G14 specifies that the acceptable levels are $|NMBE| \leq 10 \%$ and $CVRMSE \leq 30 \%$.

3.2. Component level calibration

This section examines the calibration conducted at NIST for some of the individual component models developed in the university grants. The calibration performed outside of NIST is not discussed here.

3.2.1. Chiller models

The water-cooled chiller models are user-defined *Type 8888*, modified from *Type 888*, based on *Type 666* [26]. The most relevant equations from this model are Eqs. (9) - (16). There are 13 parameters learned from the data: five in Eq. (10), five in Eq. (11), and three in Eq. (14). The variables in these equations are defined in Table 3.

$$Q_{load} = \dot{m}_{chw} c_p (T_{chw,in} - T_{chw,sp}) \quad (9)$$

$$CAP_{FL} = a_0 + a_1 T_{chw,sp} + a_2 T_{chw,sp}^2 + a_3 T_{cw,in} + a_4 T_{cw,in}^2 \quad (10)$$

$$COP_{FL} = b_0 + b_1 T_{chw,sp} + b_2 T_{chw,sp}^2 + b_3 T_{cw,in} + b_4 T_{cw,in}^2 \quad (11)$$

$$PLR = Q_{load}/CAP_{FL} \quad (12)$$

$$Power_{FL} = CAP_{FL}/COP_{FL} \quad (13)$$

$$FFLP = p0 PLR + p1 PLR^2 + p2 PLR^3 \quad (14)$$

$$Power = Power_{FL} FFLP \quad (15)$$

$$T_{chw,out} = T_{chw,in} - \frac{Q_{load}}{\dot{m}_{chw}c_p} \quad (16)$$

Chiller2 charges the TES, and the model is modified slightly in charging mode. During charging, $T_{chw,sp} = -5$ °C. As written, the model assumes that the chiller reaches this setpoint, which impacts the calculation of Q_{load} . However, in actual operations, the chiller does not reach the setpoint during charging. In general, the value of $T_{chw,out}$ is approximately 1.5 °C above the setpoint, so when the model is being used in charge mode, the setpoint is adjusted to $T_{chw,sp} = T_{chw,sp} + 1.5$ °C before the calculations shown in Eqs. (9) - (16) are carried out. If the setpoint of -5 °C is ever changed, this offset must be reassessed to see if it is still valid.

Table 3. Variable definitions for chiller equations

Name	Definition	Units
CAP_{FL}	Full load capacity	kJ/hr
COP_{FL}	Coefficient of performance at full load	
c_p	Specific heat of 30 % PG (3.48 kJ/kg-K)	kJ/kg-K
$FFLP$	Fraction of full load power	
\dot{m}_{chw}	Flow rate of PG through the evaporator	kg/hr
PLR	Part load ratio	
$Power_{FL}$	Full load power	kJ/hr
Q_{load}	Load met by the chiller	kJ/hr
$T_{chw,in}$	Temperature of the PG entering the evaporator	°C
$T_{chw,out}$	Temperature of the PG leaving the evaporator (i.e., leaving water temperature)	°C
$T_{chw,sp}$	Setpoint temperature	°C
$T_{cw,in}$	Temperature of the condensing water entering the condenser	°C

In addition, for Chiller2, two sets of parameters are used, one set using only data for the charging mode and one for general operating data. The model selects the correct set of parameters based on the setpoint temperature – if it is below 0 °C, the chiller is in charging mode, and that set of parameters is used. It should be possible to capture both modes of operation with just one set of parameters, so finding the correct set of parameters will be part of future work. For now, finding the two sets of parameters started by combining the HILFT data from multiple tests into a larger dataset. For the TES parameters, the data were limited to those where Chiller2 was on, the setpoint temperature was below zero, the leaving fluid temperature was between -2.5 °C and -5 °C, and the chiller power consumption was between 3 kW and 8 kW. For both Chiller1 and Chiller2 under normal operating conditions, the training data were limited to those where the chiller was on, and the leaving fluid temperature was within 0.56 °C of the setpoint. These restrictions ensured that the data did not include large transients that might negatively impact the optimization. For additional information about TES operations, see Sec. 4.4.

Equation (17) is the objective function J used to optimize the parameters. J is a combination of the normalized error of the power, the temperature, and the coefficient of performance (COP). The load and COP for the data are calculated as shown in Eqs. (18) - (19).

$$J = \frac{\sum_{i=1}^N |Power_i - ch_power_i|}{\max(ch_power)} + \frac{\sum_{i=1}^N |T_{chw,out,i} - ch_e_out_rtd_i|}{\max(|ch_e_out_rtd|)} + \frac{\sum_{i=1}^N |COP_i - COP_{data,i}|}{\max(COP_{data,i})} + \text{penalties} \quad (17)$$

$$Q_{load,data} = ch_f_e * c_p (ch1_e_in_rtd - ch_e_out_rtd) \quad (18)$$

$$COP_{data} = \frac{Q_{load,data}}{ch_power} \quad (19)$$

The penalty term is used to prevent non-physical solutions. This is a complex optimization problem because there are 13 parameters and many local minima, some of which yield solutions that result in negative values for capacity, COP, power, or fraction of full load power. A large penalty is added to the objective function when any of these values are negative. In addition, the parameter values were bounded, as shown in Table 4, based on observation of the equations and some initial optimizations.

Table 4. Lower and upper bounds on chiller parameters for optimization.

	a0	a1	a2	a3	a4	b0	b1	b2	b3	b4	p0	p1	p2
Lower bound	0	0	-10	-300	-10	0	-10	-10	0	0	0	0	-5000

Upper bound	200	100	10	0	10	500	10	10	50	50	200	500	0
--------------------	-----	-----	----	---	----	-----	----	----	----	----	-----	-----	---

Even with this approach, however, local minima caused significant issues. Multiple optimizations were performed using techniques from the optimization toolbox in MATLAB, including particle swarm optimization (PSO), genetic algorithm (GA), simulated annealing (SA), and multi-start *fmincon*. PSO, GA, and SA have randomness in their designs, so the results differ on each repetition when the global minimum is not found. However, these three approaches are more likely to find the global minimum than *fmincon*, and they did produce the best results. Since the results are not repeatable, dozens of optimizations were performed, and the parameters that yielded the best overall results as measured by NMBE and CVRMSE were selected and are reported in Table 5. PSO, with a swarm size of 200, produced the best parameter values. However, this result does not mean that PSO is the recommended optimization technique. It is likely that these parameter values are not at the global minimum. Future work could include a more extensive investigation of the optimization approach, perhaps considering a Pareto front since multiple valid solutions may yield the global minimum or a GA with a larger population.

Table 5. Calibrated parameter values for chiller models.

Parameter	Chiller1	Chiller2	Chiller2, Charge
a0	32.9	175.77	16.42
a1	18.5	5.82	30.25
a2	-2.5	-8.93	8.9
a3	-13	-242.93	-96.6
a4	0.87	9.81	3.47
b0	78.9	6.9	401.8
b1	-0.62	0.43	-5.68
b2	0.0087	0.0049	-6
b3	1.63	0.0027	1.97
b4	0.027	0	0
p0	46.1	7.61e-5	182.85
p1	49.7	282.29	6.7
p2	-1584.5	-4988.67	-2000

The parameter values were evaluated by using them in a TRNSYS program with just the chiller type. The datasets listed in Table 6 were combined into a single, large dataset that included 7959 data points. The output from the Chiller1 and Chiller2 models were then evaluated using (1) any data when the chiller was on and (2) data when the chiller was both on, and the chilled water temperature was within 1 °C of the setpoint. In both scenarios, the selection was based on the measured data, not the modeled data. The goal of the second scenario was to remove large transients.

Table 6. Datasets for chiller calibration.

Dataset	Variation
2022_08_09_run1	Typical summer day, shift with TES
2022_05_05_run2	Extreme summer day, eff
2022_05_10_run4	Typical summer day, eff
2022_04_01_run1	Typical summer day, shed
2022_08_31_run3	Typical summer day, shift
2022_05_12_run4	Extreme summer day, shed
2022_09_19_run4	Extreme summer day, shift

The overall accuracy of the models is shown in Fig. 9 (note: the Chiller2 results include charging data). The temperature difference used for the NMBE and CVRMSE calculations is $T_{chw,out} - T_{chw,in}$. The “sub” subscripts are the results with the transient data removed, and the grey region encompasses the acceptance criteria from G14. In general, removing transient data improves the results, bringing them within the bounds of the acceptance criteria. The RMSE of $T_{chw,out}$ is less than 1 °C for each dataset.

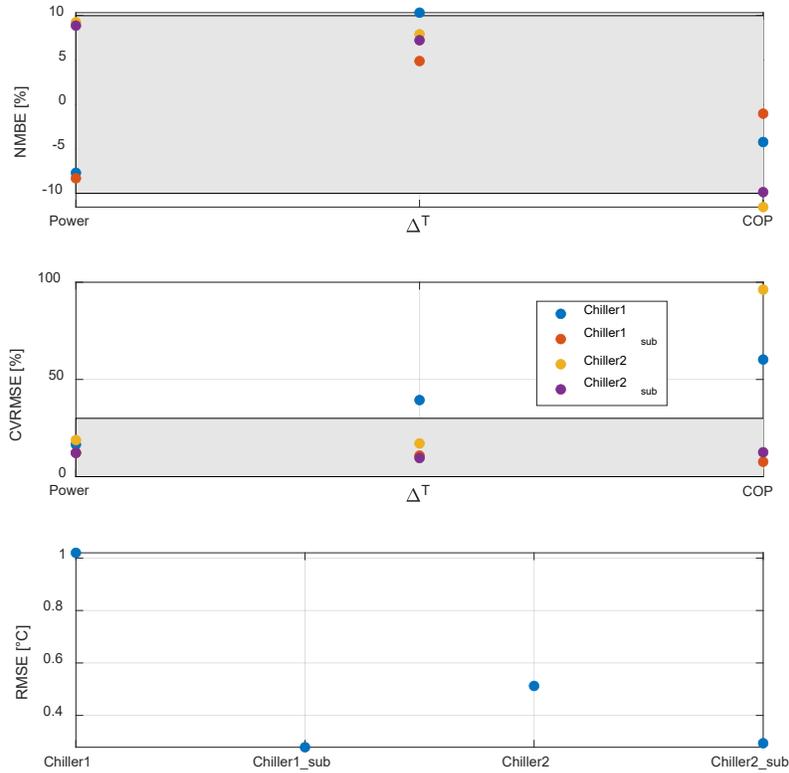


Figure 9 Chiller model results.

Figures 10 and 11 show the Chiller1 and Chiller2 data and simulation results for power and chilled water temperature for timesteps when the chiller is enabled ($ch_on > 0$) and when the large transients are removed, respectively.

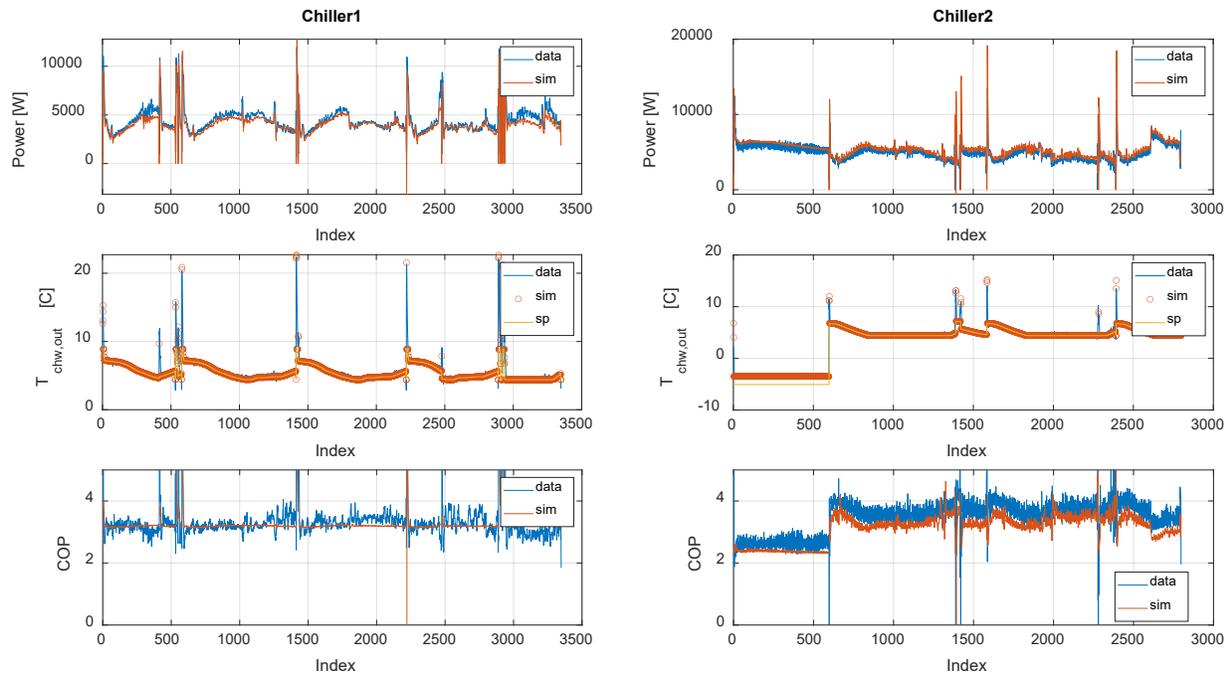


Figure 10 Comparison between data and simulation for chiller power and chilled water temperature.

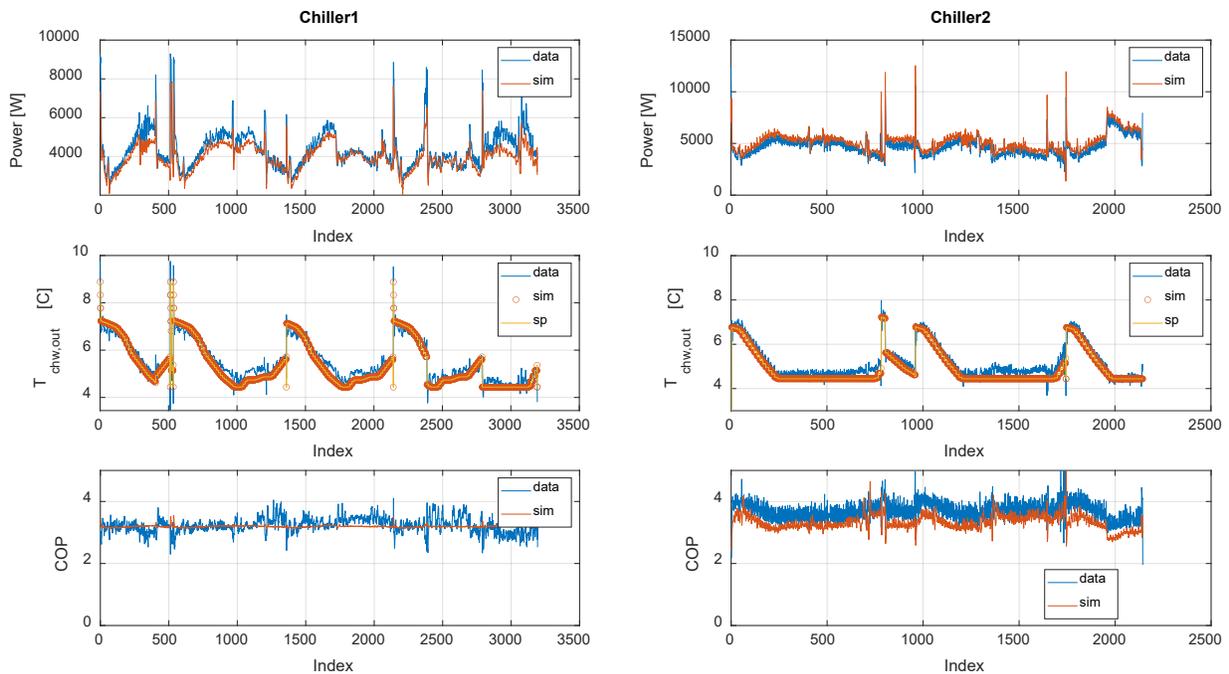


Figure 11 Comparison between data and simulation for chiller power and chilled water temperature with large transients removed.

There are several causes for the discrepancy between the model and the data. Figure 12 shows a section of the Chiller1 data highlighting one of the transient periods, in this case, caused by the chiller cycling. During this period, the chiller is enabled, so $ch1_on = 1$, but the load is low, so the chiller turns off until $T_{chw,out}$ exceeds the setpoint by $1.7\text{ }^{\circ}\text{C}$ ($3\text{ }^{\circ}\text{F}$), at which point it turns back on. The model has difficulty capturing this behavior because there is limited cycling data in training, and it assumes that if the chiller is not on, the power is zero. However, even when the chiller is off, there is a small non-zero power draw ($< 50\text{ W}$).

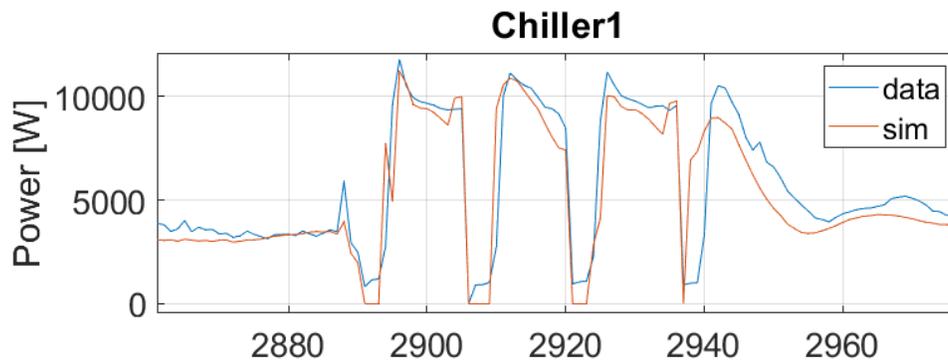


Figure 12 Close-up of a period of cycling in Chiller1.

The plots in the middle row of Figure 11 show the temperature results when the large transients are removed, and it is apparent that the actual temperature is above the setpoint for much of the data. Part of this disparity is that the internal chiller control holds the setpoint to within $0.28\text{ }^{\circ}\text{C}$ ($0.5\text{ }^{\circ}\text{F}$), and the measurement of the chiller temperature has an uncertainty of about $0.11\text{ }^{\circ}\text{C}$ ($0.2\text{ }^{\circ}\text{F}$). In addition, the chiller control sets the compressor operation by comparing the internal temperature measurement with the setpoint. That internal measurement is different from the laboratory measurement, in some cases by as much as $0.22\text{ }^{\circ}\text{C}$ ($0.4\text{ }^{\circ}\text{F}$), but on average, the difference is around $0.11\text{ }^{\circ}\text{C}$. These factors together mean that the actual $T_{chw,out}$ can differ from the setpoint by approximately $0.31\text{ }^{\circ}\text{C}$, but the model assumes they are equal. For additional context, this difference is approximately 10 % of the temperature change across the evaporator.

For Chiller1, the modeled COP (lower left of Figure 11) is nearly constant, with only small fluctuations. This outcome reflects the narrow range of loads in these data. When power spikes occur, the COP spikes as it does in the data. For Chiller2, the modeled COP (lower right of Figure 11) follows the same trend as the data but underpredicts. This result occurs because the chiller power is overpredicted, resulting in a larger denominator and, therefore, smaller COP. The region where the COP is nearly constant is when the chiller was charging the TES and the system operated at steady state.

These models have sufficient accuracy for the current study based on the metrics defined in Sec. 3.1, but improvements could be made by:

- making a more extensive selection of an optimization algorithm and its settings,
- using more data to optimize the parameters, and
- using a different modeling approach, such as that presented in [3], or a quasi-steady model that includes heat capacities and other time constants.

3.2.2. Hydronic flow, pressure, and pump power

Type 9031 calculates the flow rates and pressures in the hydronic system and the pump power consumption. It was developed based on *Type 9000* from grant 70NANB21H108. The bridge between the PL and SL allows for the pressures and flows in those loops to be solved separately. Although this type determines the pressures, they are unused right now and are not calibrated or validated in this study.

3.2.2.1. Overview of the flow calculations

This section provides an overview of the procedure for calculating the flow rates in the SL and PL. The following section presents the equations related to the flow calculations with parameters that need to be determined through calibration. The components in this section can be found in Fig. 3.

First, consider the SL. Figure 13 is the pseudocode used to calculate the SL flow rate, Q_{SL} , and the flow rates through each cooling coil (Q_{AHU1} and Q_{AHU2}) and HX1 (Q_{HX1}). The inputs to this type include the speed of Pump3, N_{P3} , the position of the cooling coil valves, $V12$ and $V13$, and the position of the valve for HX1, $V10$. The low pressure in the system is assumed to be 55 kPa (8 psi); this value is the pressure upstream of Pump3 and downstream of the valves. The pressure drops through the cooling coils and HX1 are not explicitly modeled but are instead lumped into the estimation of the pressure drop across the valve. Figure 14 is the pseudocode for calculating the flow rate through the valves. This iterative calculation guesses the flow rate through each valve and then calculates the pressure drop through that valve. This approach is the logic implemented in *Type 9000*.

```
Secondary Loop Calculations: Determine SL flows and pressures.  
  
0: Inputs: N_P3  
1: Initial values: Q_SL = 30; err = 999; tol = 0.01; it = 0  
2: while err > tol && it < 1000  
3:   Given Q_SL and N_P3, calculate pressure rise across pump  
4:   Given pressure rise across pump, calculate the pressure upstream of  
   the three valves in the SL  
5:   Given the valve positions and the pressure drop across the valves,  
   calculate the flow rate through each valve - Q_V(V,DP)  
6:   Set Q_SL_c to the sum of the flow rates through the three valves  
7:   Calculate the normalized error between Q_SL and Q_SL_c  
8:   Set Q_SL = Q_SL_c  
9:   it = it + 1  
10: end while  
11: return Q_valves, Q_SL
```

Figure 13 Pseudocode describing the process to calculate the flow rates in the SL.

```
Flow Through Valves: Determine flow through a valve.  
  
Q_V(V, DP)  
1: Initial values: Q = 15; err = 999; tol = 0.001; it = 0  
2: while err > tol && it < 100  
3:   Given V and Q, calculate pressure drop across valve (DP_c)  
4:   Given V and Q+0.01Q, calculate pressure drop across the valve (DP_cp)  
5:   Calculate dDPdQ = (DP_cp-DP_c)/0.01  
6:   Update Q = Q+(DP-DP_c)/dDPdQ  
7:   Calculate the normalized error, err, between DP_c and DP  
8:   it = it + 1  
9: end while  
10: return Q_valve
```

Figure 14 Pseudocode describing the process to calculate the flow rates in the SL.

The *Type 9000* flow calculations in the PL did not match the measured HILFT data well, leading to the development of *Type 9031*. In this type, the flow rates are hardcoded. If Chiller1 operates, Pump1 is used, and the flow rate is 4.77 m³/hr (21 gpm). If Chiller2 or the TES are in use, Pump2 is operating. If Chiller2 is used to meet the building load, then the flow rate is set to 5.22 m³/hr (23 gpm). The flow rate is 4.77 m³/hr if the TES is in use. The overall flow rate of the PL is then set to either the Pump1 or Pump2 flow rate. In the TES discharge mode, the flow rate

is split between the bypass around the TES and the flow through the TES. Figure 15 is the pseudocode for determining flow rates in the PL.

```
Primary Loop Flows: Determine flow rates in the PL

PrimaryLoop(Q_PL, V8, V18, V19, N_P1, N_P2)
1: if V18 > 0 and V8 > 0
2:   Q_C2 = 4912           // some or all flow goes through the TES
3: elseif V18 > 0
4:   Q_C2 = 5380         // Chiller2 is in use, but the TES is bypassed
5: else
6:   Q_C2 = 0
7: endif
8: if V19 > 0
9:   Q_C1 = 4912         // Chiller1 is in use
10: else
11:   Q_C1 = 0
12: endif
13: Q_PL = Q_C1 + Q_C2
14: if V8 = 1           // all flow goes through the TES
15:   Q_TES = Q_PL
16:   Q_BTES = 0
17: elseif V8 = 0       // all flow goes through the TES bypass
18:   Q_TES = 0
19:   Q_BTES = Q_PL
20: elseif V8 <= 0.5    // flow is split between the TES and bypass
21:   Q_TES = Q_PL/2; err = 999; tol = 0.01; it = 0
22:   while err > tol && it < 100
23:     Calculate the pressure drop across V8
24:     Calculate Q_BTES using the parameters for 81 in Table 8
25:     Calculate Q_TES_c = Q_PL - Q_BTES
26:     Calculate the normalized, err, error between Q_TES and Q_TES_c
27:     Q_TES = Q_TES_c
28:     it = it + 1
29:   end
30: endif
31: return Q_PL, Q_BTES, Q_TES, Q_C1, Q_C2
```

Figure 15 Pseudocode for calculating flows in the PL.

Type 9000 did not work well primarily because it uses the pressure drops in the system to calculate the flow rate. This approach is appropriate, but the pressure drop data in the IBAL are

insufficient to calibrate the parameters properly. Future work will look for an alternative to hardcoding the flow rates, allowing for the modeling of more energy-efficient variable-speed control of the pumps. A straightforward option is to use data to find a factor that scales the flow rate as a function of pump speed. Although this method would not capture the effect of variable system pressure on the flow rate, it may be sufficient for typical operating conditions.

3.2.2.2. Parameters for the flow calculations

This section presents the equations used to calculate the flow rates with parameters calibrated from data. For completeness, the parameter values for the PL are included here, but as discussed in the previous section, except for the parameters for V8, these values are not used in *Type 9031*. The pressure drop through a valve is calculated using Eqs. (20) – (21). The parameters in these equations, $a0$ and $a1$, are calibrated from data for each valve. In these equations, Q is the volumetric flow rate in gpm, and V is the valve position as a fraction from 0 to 1. The resulting pressure drop is in psi.

$$Q_{ratio} = \frac{Q}{a0 V} \quad (20)$$

$$DP_{valve} = a1 Q_{ratio}^2 \quad (21)$$

The pressure rise across the pump is calculated using Eqs. (22) – (26); each pump has eight parameters learned from data. The pump speed is in rpm, the flow is in gpm, and the pressure is in psi.

$$DP_{max} = a2 + a3 N + a4 N^2 \quad (22)$$

$$Q_{max} = a5 + a6 N \quad (23)$$

$$Q_{ratio} = Q/Q_{max} \quad (24)$$

$$DP_{ratio} = a7 + a8 Q_{ratio} + a9 Q_{ratio}^2 \quad (25)$$

$$DP = DP_{max} DP_{ratio} \quad (26)$$

Specific tests were run as part of grant 70NANB21H108 to generate the data used to determine the parameter values. The final values are presented in Table 7 and Table 8.

There are three valves in the SL, V10, V12, and V13, and three in the PL, V19, V18, and V8. V8 operates as a three-way valve, so the pressure drop is calculated for two scenarios. In the first scenario, 81, the valve allows flow to bypass the TES, and in the second scenario, 82, the valve allows some or all the flow to go through the TES. The valve parameters are listed in Table 7. Each valve has two parameters, $a0$ and $a1$. Table 8 contains the parameter values for the pumps. As mentioned previously, in *Type 9031*, the pressure drops across valves V18 and V19 are not used to calculate the flow rates.

Table 7. Parameter values for the valves in *Type 9031*.

Component	Parameter values, $a0$ and $a1$
V8, bypass	20, 11
V8, through TES	20, 11
V10	39, 19
V12	7.2, 19
V13	19, 22
V18	22, 25
V19	22, 25
V20	22, 25

Table 8. Parameter values for the pressure rise across the pumps in *Type 9031*.

Component	Parameter values, $a2$ through $a9$
Pump1	-0.0156, -4E-5, 8E-6, 0.0, 0.0346, 1.0, 0.0194, -0.4183
Pump2	0.0, 0.0006, 2E-6, -10.501, 0.0184, 1.0, -0.0494, -0.5643
Pump3	-0.0002, 1E-7, 7E-6, 0.0, 0.0328, 1.0, 0.0021, -0.4177
Pump5	0, -0.0005, 6E-06, -11.2, 0.035, 1.0, 0.091, -0.46

3.2.2.3. Pump power consumption

The equations for the pump power consumption calculation are Eqs. (27)—(31). They are like those for the pressure rise across a pump, but the parameter values differ. The pump speed, N , is in rpm, the flow rate is in gpm, and the power is in W.

$$Power_{max} = a10 + a11 N + a12 N^2 \quad (27)$$

$$Q_{max} = a13 + a14 N \quad (28)$$

$$Q_{ratio} = Q/Q_{max} \quad (29)$$

$$Power_{ratio} = a15 + a16 Q_{ratio} \quad (30)$$

$$Power = Power_{max} Power_{ratio} \quad (31)$$

Table 9 contains the parameter values for the pump power; the *Calibration* column indicates if the value is from the grant or calibrated as part of the current work at NIST. For Pump1 and Pump2, the values were found using a genetic algorithm in MATLAB with a population size of 500 and a selection function of *selectionroulette*. For Pump3 the population size was 1000 with a selection function of *selectiontournament*.

Table 9. Parameter values for the pump power in Type 9031.

Component	Calibration	Parameter values, <i>a10</i> through <i>a16</i>
Pump1	NIST	192.37, -2.75, 0.014, 129.25, 8.82, 0.0165, 4.36
Pump2	NIST	0.82, -20.5, 0.019, 66.43, 6.88, 0.011, 20.20
Pump3	NIST	101.7, -9.23, 0.011, -9.63, -5.02, 0.031, -5.3
Pump5	grant	153.7, -0.38, 0.0004, -11.2, 0.035, 0.78, 0.27

3.2.2.4. Validation

Figure 16 shows the performance metrics for the pump power calculations. NMBE and CVRMSE pass the thresholds defined by G14 for all three pumps. Figure 17 shows each pump's time series IBASIM (*sim*) and HILFT (*data*) profiles. The Pump1 model underpredicts the data, but the Pump2 and Pump3 models are closer. The Pump1 and Pump2 models predict a constant value for power because although they are variable speed pumps, they operate at constant speed and with limited variability in valve operations. Pump3 operates at different speeds, and although the model does not include any component directly related to pressure, the speed of Pump3 modulates to meet a pressure setpoint, so the change in speed captures the effects of pressure, and the model captures the slight variability in the data. However, it has been validated over a limited range of operating conditions, so it should be re-evaluated if conditions change.

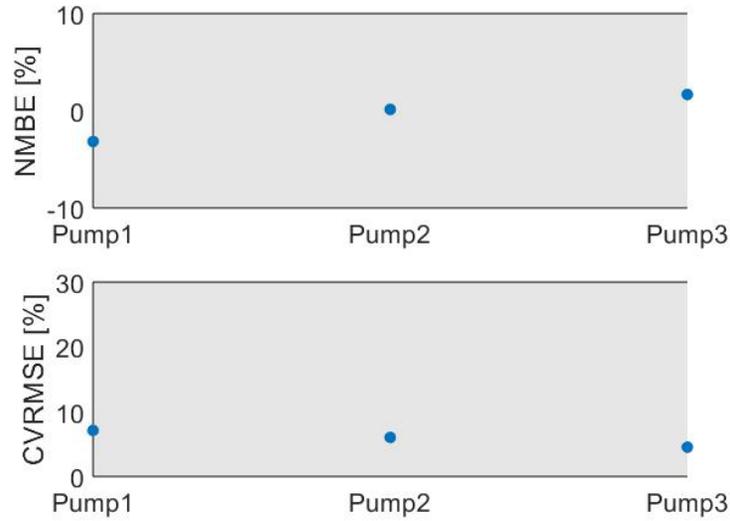


Figure 16 Pump power results.

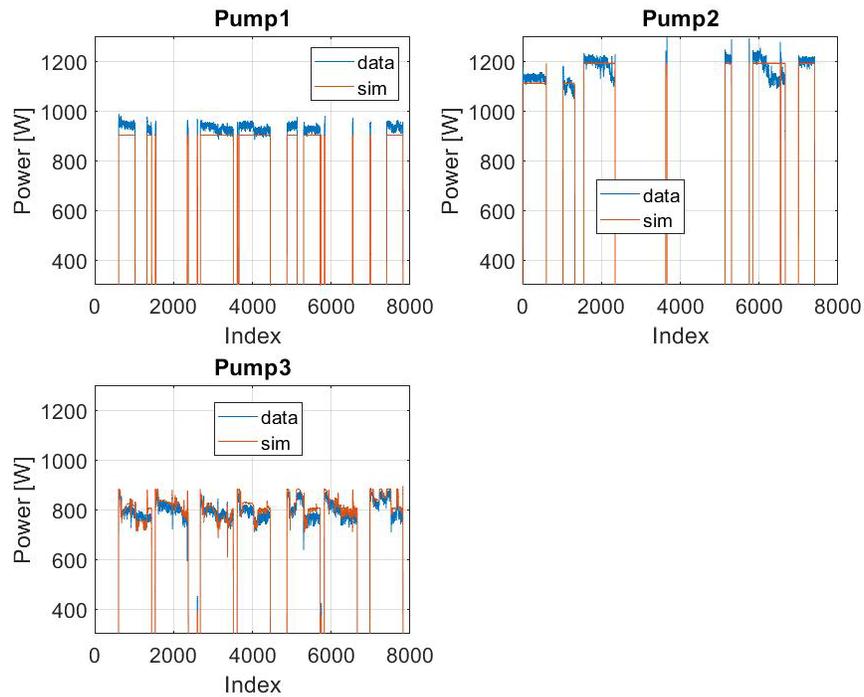


Figure 17 Comparison of the simulated and measured values for the pump powers.

The flow rate model is integrated with the system, so validation is only performed at the system level, as discussed in Sec. 5. The pump power model is also validated at the system level, and those results will be discussed in that section.

3.2.3. AHU fan power consumption

The AHU fan power consumption is calculated using a decision tree implemented in MATLAB. Ideally, fan power is calculated based on airflow rate and fan speed. However, the fan speed is difficult to model correctly, so instead, a decision tree was created that uses the static pressure setpoint, the outdoor airflow setpoint, the VAV airflow rates, and the errors between the zone temperature and the zone temperature setpoint for the two zones served by the AHU. In practice, the fan speed is calculated from a proportional-integral (PI) control loop that tries to control the static pressure downstream of the fan to a setpoint calculated in HSIM [27]. The static pressure is a function of the damper positions in the VAVs. The damper positions are based on a PI loop that tries to control the airflow rate through the VAV to an airflow rate setpoint (see Sec. 4.6). The airflow rate setpoint is based on a PI loop that tries to control the zone temperature to the setpoint from HSIM. So, the variables used by the decision tree are based on the variables relevant to this multi-layer control scheme. Although different settings for the tree models were evaluated, the default settings for the Fine Tree model in MATLAB yielded the best results. The tree models were built using the datasets in Table 6. These datasets all use a simulated building located in Atlanta, GA, US. The models were validated using the datasets in Table 10, which all use a simulated building in Tucson, AZ, US.

Table 10. Datasets for validating the AHU fan power models.

Dataset	Variation
2023_04_14_run4	Typical summer day, shift with TES
2022_05_09_run4	Extreme summer day, eff
2022_03_23_run3	Typical summer day, eff
2022_04_06_run2	Typical summer day, shed
2022_09_12_run1	Typical summer day, shift
2022_03_01_run6	Extreme summer day, shed
2022_09_16_run1	Extreme summer day, shift

The performance metrics for the tree models for each AHU fan are presented in Fig. 18. The fan model meets all the thresholds from G14; however, since the CVRMSE are both over 20 %, an improvement to these models will be included in future work.

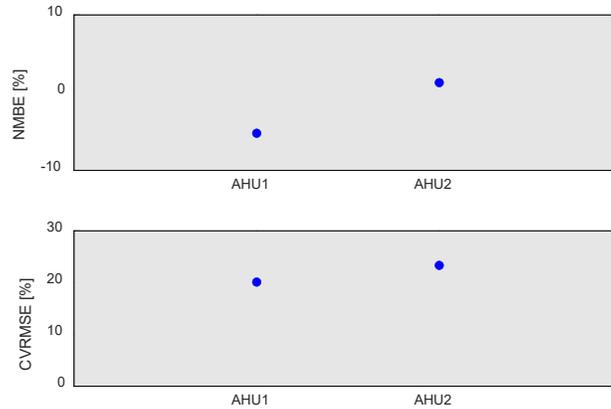


Figure 18 Results for AHU fan power consumption model.

3.2.4. AHU cooling coil

The cooling coils (CCs) in the AHUs are modeled by *Type 9898*, which is a slight variation on *Type 9897*, developed in grant 70NANB21H108. The model uses a bypass approach in which a portion of the air bypasses the coil while the rest passes through the coil. The two streams then mix downstream to determine the final air properties. In the case of the IBAL, based on calibrations to laboratory data, the bypass fraction is 3 %. The model of the coil takes the coil capacitance into account so that the model includes an approximation of the dynamic behavior of a coil. The details of this model can be found in Ref. [11]. *Type 9897* does not include the effect of fan heating, so the model was modified to include an enthalpy term representing fan heating. The key parameters used in the CC models for both AHUs are listed in Table 11; the origin of the values is in the “source” column. Although the two coils are nominally identical, their parameter values differ. This discrepancy may be due to differences in the range of operation of each CC in terms of airflow rate, liquid flow rate, inlet air temperature and humidity, and fan speed. These differences could yield different values for the overall heat transfer coefficient times area (UA) and fan heating. This finding could also indicate that the model is too simple and does not adequately capture the physics of the system.

Table 11. Parameters in the cooling coil models for the AHUs.

Parameter	AHU1	AHU2	Source
Coil capacitance	47.5 kJ/K	6.4 kJ/K	NIST
Bypass fraction	0.03	0.03	Grant
UA	8311 kJ/hr-K	10566 kJ/hr-K	NIST
Fan heating	1.2 kJ/kg	1.03 kJ/kg	NIST

The cooling coil type was validated using the datasets from Table 6. In this case, the cooling coil is not validated in isolation. Instead, the zone temperatures and humidity ratios from HSIM and the VAV airflow rate setpoints from ISIM are supplied to a mixer that determines the return air temperature and humidity. The return airflow is calculated as the difference between the sum

of the VAV airflows and the outdoor airflow rate supplied by HSIM. HSIM also provides the outdoor air temperature and humidity; another mixer calculates the inlet air temperature and humidity entering the cooling coil. The temperature of the PG entering the cooling coil, $T_{in,pg}$, and the PG flow rate are from the data. The metrics shown in Fig. 19 are:

- $\Delta T_{air} = T_{out} - T_{in}$
 - T_{out} - dry bulb temperature downstream of the cooling coil, calculated by *Type 9898*,
 - T_{in} - dry bulb temperature upstream of the cooling coil, calculated outside of *Type 9898*,
- RH_{out} – relative humidity downstream of the cooling coil, calculated by *Type 9898*,
- RH_{in} – relative humidity upstream of the cooling coil, calculated outside of *Type 9898*,
- $\Delta T_{liq} = T_{out,pg} - T_{in,pg}$
 - $T_{out,pg}$ – temperature of the PG leaving the cooling coil, calculated by *Type 9898*,
 - $T_{in,pg}$ – temperature of the PG entering the cooling coil, input to the model, and
- q_{liq} – liquid side heat transfer.

As apparent from the number of points outside the grey area, these systems have mixed results relative to the G14 thresholds. *Type 9897* was tested in isolation, but the results were similar. The impact of the inaccuracies in the CC model is discussed in Sec. 5.

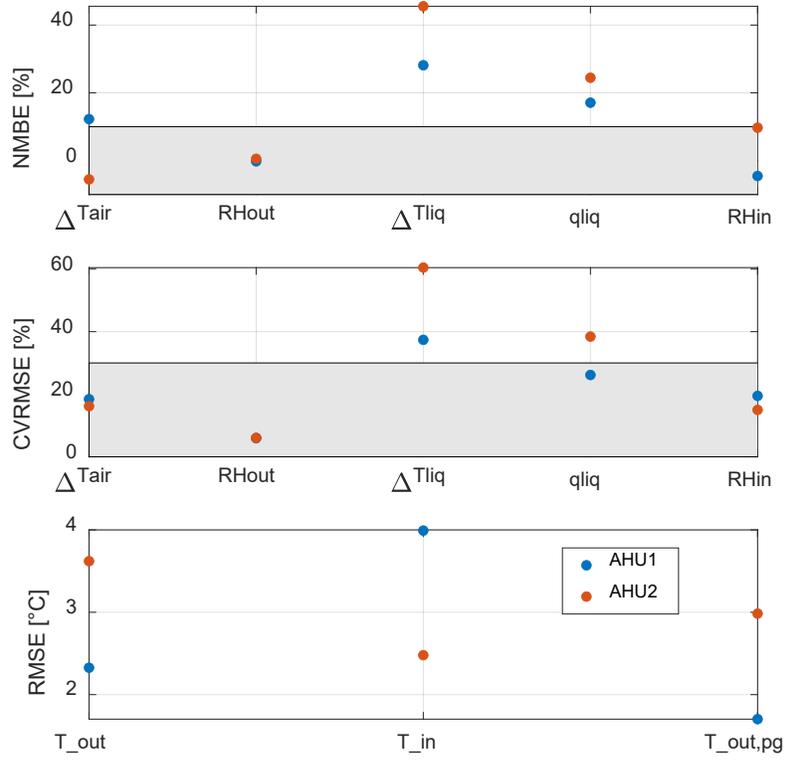


Figure 19 Results for AHU cooling coil models.

4. Added Controllers

As mentioned in Sec. 2.3, TSIM (the TRNSYS part of IBASIM) lacked several controllers, and some controllers were not sufficiently accurate when compared to actual laboratory test data, so new controllers were developed and implemented in MATLAB. This section introduces these controllers and highlights which ones might be replaced by more intelligent approaches in the future. These controllers are adapted from the ones used in the IBAL and detailed in NIST Technical Note 2178 (TN2178) [27]. TN2178 provides an overview of the controls in a commercial building, so many of the details of the controls are not repeated here.

4.1. Valve controllers

Three valves in the IBAL need to be explicitly modeled because they determine flow rates in the hydronic system. The first is V8, the valve that determines the flow rate through the TES and the bypass around the TES. If $V8 = 0$, all the flow bypasses the TES; if $V8 = 1$, all flow goes through the TES. Anything in between means that the TES is being discharged, and the flow rate through the TES is based on a PI controller that modulates V8 to control the temperature leaving the PL, pl_out_rtd (see Fig. 2), to a setpoint. The PI logic is implemented in MATLAB; the settings are shown in Table 12 and are the same as those used in the IBAL. TN2178 shows the complete set of equations describing the implementation of the PI logic.

The other two valves, V12 and V13, are for the cooling coils for AHU1 and AHU2, respectively. Each of these valves modulates to control the supply air temperature at the exit of each AHU, $ahu1_out_rtd$ and $ahu2_out_rtd$, to the setpoint (see Fig. 5). The PI parameters, shown in Table 12, differ from what is used in the IBAL. When the gains (K_p and K_i) from the IBAL were used in ISIM (the TRNSYS and MATLAB models of the IBAL in IBASIM), the valves were unstable, so the gains were re-tuned. The bias term, u_{Bias} , for the CC valve controllers is variable. The actual valves are non-linear, so to generate the correct behavior in the system, an algorithm calculates a bias term that can cause a larger position change for the valve when necessary. The pseudocode for this operation is shown in Fig. 20.

Table 12. Settings for the PI logic for valves.

Name	V8	AHU1	AHU2	Definition
K_p	0.5	0.15	0.15	Proportional gain
K_i	0.013	0.001	0.001	Integral gain
SP_{High}	18.3 °C (65 °F)	21.1 °C (70 °F)	21.1 °C (70 °F)	Maximum value of the setpoint
SP_{Low}	-1.1 °C (30 °F)	-1.1 °C (30 °F)	-1.1 °C (30 °F)	Minimum value of the setpoint
u_{Bias}	0.5	variable	variable	Bias term
A	-1	-1	-1	Action (direct = 1, reverse = -1)
EU_{High}	9.5 V	8 V	8 V	Maximum value of the control signal in engineering units
EU_{Low}	0 V	2 V	2 V	Minimum value of the control signal in engineering units

The basic concept for determining that a bias is needed is to see if the actual air temperature, T_{SA} , is different from the setpoint, $T_{SA,SP}$, by more than a threshold, $thresh$ (all values are in °F), for N timesteps (in minutes). If T_{SA} is greater than the setpoint, there is a need for more cooling, so the bias term increases by an increment of inc . This increment is based on a valve position between 0 and 1, where, for the CC valves, 0 corresponds to 2 V and 1 corresponds to 8 V. In this pseudocode, V_{CC} is the valve position in volts. If the valve position is near the minimum or maximum, then u_{Bias} remains unchanged.

Cooling Coil Valve Bias: Determine the magnitude of the bias term for the cooling coil valves

ccBias(T_{in} , T_{SA} , $T_{SA,SP}$, V_{CC})

```
0: N = 5; thresh = 0.5; inc = 0.01; count = 0; high = 8; low = 2
1: if  $T_{in} \geq T_{SA} + 2$ 
2:   delta = 0
3: else
4:   delta =  $T_{SA} - T_{SA,SP}$ 
5: endif
6: if  $abs(delta) \leq thresh$ 
7:   count = 0
8: else
9:   count = count + 1
10: endif
11: if count  $\geq N$ 
12:   count = 0
13:   if ( $delta > 0$ ) and ( $V_{CC} < high - 1$ ) // open the valve more
14:      $u_{Bias} = u_{Bias} + inc$ 
15:   elseif ( $delta < 0$ ) and ( $V_{CC} > low + 1$ ) // close the valve more
16:      $u_{Bias} = u_{Bias} - inc$ 
17:   endif
18: endif
19: store  $u_{Bias}$ , count
20: return  $u_{Bias}$ 
```

Figure 20 Pseudocode for setting the bias term for the cooling coil valves.

4.2. Secondary loop operation

This section focuses on the decision to turn the SL on. The SL is the connection between the hydronic and air systems. If the air system requests cooling, the SL pump will turn on, and the PL will turn on in response to the SL. The basic logic is shown in Fig. 21. In the IBAL and in ISIM, a request for cooling is determined by looking at the average cooling coil valve positions,

$V12Avg$ and $V13Avg$, during a specified time frame. In ISIM, that time frame is two minutes; in the IBAL, that time frame is 90 seconds. If either CC valve is open on average (i.e., $V12Avg$ or $V13Avg > 2$ V) and there is airflow ($ftotal > 100$ cfm) during the two-minute time frame, then there is a call for cooling and the SL pump will turn on. Once the SL pump is on, that is an indication that the PL should also turn on to provide cooling.

SL Controller: Determine if the SL should be turned on
<u>slController(V12,V13,ftotal)</u> 0: $N = 2$; $count = 1$; $V12Sum = V12$; $V13Sum = V13$; $ftotalSum = ftotal$; $cool = 0$ 1: $count = count + 1$ 2: $V12Sum = V12Sum + V12$ 3: $V13Sum = V13Sum + V13$ 4: $ftotalSum = ftotalSum + ftotal$ 5: $V12Avg = V12Sum / count$ 6: $V13Avg = V13Sum / count$ 7: $ftotalAvg = ftotalSum / count$ 8: if $count \geq N$ 9: if ($ftotalAvg > 100$) and ($V12Avg > 2$ or $V13Avg > 2$) 10: $cool = 1$ 11: else 12: $cool = 0$ 13: endif 14: $V12Sum = V12$ 15: $V13Sum = V13$ 16: $ftotalSum = ftotal$ 17: endif 18: return $cool$

Figure 21 Pseudocode for determining if there is a call for cooling that requires the secondary loop to operate.

4.3. Chiller staging

The IBAL has two chillers: Chiller1 is nominally 26.4 kW (7.5 tons), and Chiller2 is nominally 35 kW (10 tons). One important control decision is when to use each chiller. In the IBAL, the staging decisions are based in part on the chiller power consumption, but that approach does not work well with the simulation because the transient power consumption is not modeled well. For the baseline staging in the simulation, the decision is based instead on the building load. Figures 22 and 23 show the pseudocode for the chiller staging. It was tuned to provide results similar to the IBAL but does not match perfectly. At this point, Chiller2 is used to charge the TES, but that is a decision that an advanced control approach could determine.

Currently, there are three stages:

- stage = 0 – no chiller is enabled,
- stage = 1 – Chiller1 is enabled, and
- stage = 2 – Chiller2 is enabled.

When a test first starts, the stage is 0. If the building load exceeds $q1_{min} = 2$ kW for more than $n_{viol} = 8$ minutes, the stage becomes 1, and Chiller1 is enabled. There is also a transition period of 1 minute. During that time, the stage cannot change. In the IBAL, this transition is 10 minutes because when the chiller is enabled, it does not immediately turn on, and once it does turn on, it ramps up slowly. The transition time is unnecessary since the model does not capture these behaviors. If the initial load exceeds $q2_{min} = 13$ kW, the stage is 2, and Chiller2 is enabled. In general, this scenario does not happen. The typical operation is stage 0 to stage 1 to stage 2 if the load becomes high enough.

Once the system is in stage 2, Chiller2 will stage down if the load is less than $q2_{sd} = 10$ kW. At this point, the stage becomes 1, and Chiller1 will turn on. Chiller1 stages down if the load is less than $q1_{sd} = 1$ kW, and then the stage = 0. Both chillers have a variable frequency drive on the compressor, so they can each be used for a wide range of loads, and their operating ranges overlap substantially. A more intelligent control approach would be to consider which chiller is more efficient for a given load and setpoint requirement while also considering the run time on each chiller, which can impact longevity. Such an approach will be investigated in future work, while the current staging approach serves as a baseline for comparison.

Chiller Staging: Determine the chiller stage
ChillerStaging(ch1_{power},ch2_{power},T_{chwst},T_{chw},load)
0: n_viol = 8; n_trans = 1 initialize stage = 0; N = 0; trans01 = trans02 = trans12 = trans21 = 0
1: call chillerLoadViolation (stage,ch1 _{power} ,ch2 _{power} ,T _{chwst} , load)
2: if load <= 0
3: stage = 0; N = 0; trans01 = trans02 = trans12 = trans21 = 0
4: elseif trans01+trans02+trans12+trans21 > 0
5: in transition between stages, do not change the stage
6: N = N + 1
7: elseif stage == 0
8: if n02 > n_viol
9: stage = 2; trans02 = 1;
10: elseif n01 > n_viol
11: stage = 1; trans01 = 1;
12: else
13: stage = 0;
14: endif
15: elseif stage == 1
16: if n12 > n_viol
17: stage = 2; trans12 = 1;
18: elseif n10 > n_viol
19: stage = 0;
20: else
21: stage = 1;
22: endif
23: elseif stage == 2
24: if n21 > n_viol
25: stage = 1; trans21 = 1;
26: else
27: stage = 2;
28: endif
29: else
30: stage = 0; N = 0; trans01 = trans02 = trans12 = trans21 = 0
31: endif
32: if N >= n_trans
33: N = 0; trans01 = trans02 = trans12 = trans21 = 0
34: endif
35: if stage == 1
36: ch1On = 1; ch2On = 0
37: elseif stage == 2
38: ch1On = 0; ch2On = 1
39: endif
40: return stage, ch1On, ch2On

Figure 22 Pseudocode for chiller staging.

<p>Load Violation: Determine if a chiller is needed and if so, which chiller</p> <pre> chillerLoadViolation(stage,ch1_{power},ch2_{power},T_{chwst}, load) 0: q2_min = 13; q2_sd = 10; q1_min = 2; q1_sd = 1 initialize n01 = n02 = n10 = n12 = n21 1: if stage == 0 2: if load >= q2_min // building load exceeds the minimum turn on Chiller2 3: n02 = n02 + 1; n01 = n12 = n10 = n21 = 0 4: elseif load > q1_min // building load exceeds the minimum turn on Chiller1 5: n01 = n01 + 1; n02 = n12 = n10 = n21 = 0 6: else 7: n01 = n02 = 0 // stay in stage = 0 8: endif 9: elseif stage == 1 10: if load >= q2_min // building load exceeds the minimum turn on Chiller2 11: n12 = n12 + 1; n01 = n02 = n10 = n21 = 0 12: elseif load < q1_sd // building load is less than the minimum to turn Chiller1 off 13: n01 = n01 + 1; n02 = n12 = n10 = n21 = 0 14: else 15: n12 = n10 = 0 // stay in stage = 1 16: endif 17: elseif stage == 2 18: if load < q2_sd // building load is less than the minimum to turn Chiller2 off 19: n21 = n21 + 1; n01 = n02 = n10 = n12 = 0 20: else 21: n21 = 0 // stay in stage = 2 22: endif 23: endif 24: store n01, n02, n10, n12, n21 25: return n01, n02, n10, n12, n21 </pre>
--

Figure 23 Pseudocode for determining if the chiller stage should be changed.

4.4. TES operation

The TES has three modes of operation: bypass, charge, and discharge. Bypass mode means that the TES is unused; all the flow goes around the TES. In charge mode, Chiller2 operates with a setpoint of -5 °C to build ice. In discharge mode, the TES provides cooling for the air system. The amount of PG that flows through the TES is controlled by modulating V8 (see Fig. 2) using a PI loop. The goal is to move pl_out_rtd towards the chilled water temperature setpoint. In discharge mode, the TES can be the sole source of cooling, but it can also be used in coordination with a chiller.

In HSIM (the virtual building model and controllers), the TES operation is set based on occupancy and electric utility rates. The TES is in charge mode after occupancy ends at 20:00 until occupancy starts the following day at 06:00. In discharge mode, the TES is the sole source of cooling, and the discharge period corresponds to the time of peak electric rates.

4.5. Chilled water temperature setpoint

The baseline approach for calculating the chilled water temperature setpoint, T_{chwst} , which is used by the chiller, SL, and TES in discharge mode, is based on the outdoor air temperature, T_{OA} , and is shown in Fig. 24. Every 15 minutes, T_{chwst} is recalculated based on the mean T_{OA} from the prior 15 minutes. If the T_{OA} is above 32 °C or below 15.6 °C, T_{chwst} is set to a constant value.

Chilled Water Temperature Setpoint: Determine the chilled water temperature setpoint used by both the chiller and the secondary loop

tchwst(T_{OA})

```
0: n = 14; N = 0
   Tchwst0 = max[min(-0.2671 TOA + 13.0462, 8.9), 4.4]
   Tchwst,t-1 = Tchwst0
   TOA,hist = [0]*n
1: N = N + 1
2: if N <= n
3:   Tchwst = Tchwst,t-1
4:   TOA,hist[N] = TOA
5: else
6:   TOA,hist[N] = TOA
7:   TOA = mean(TOA,hist)
8:   if TOA > 32.2
9:     Tchwst = 4.4
10:  elseif TOA < 15.6
11:    Tchwst = 8.9
12:  else
13:    Tchwst = -0.74*TOA + 17.8
14:  endif
15: Tchwst,t-1 = Tchwst
16: N = 0
17: TOA,hist = [0]*n
18: endif
19: return Tchwst
```

Figure 24 Pseudocode for determining the chilled water temperature setpoint, which is used by the chiller and the SL.

4.6. Airflow setpoint

In the IBAL, the airflow rate in the VAVs is set using a dual maximum reset scheme [28, 29]. This scheme sets separate maximum airflow rates for cooling and heating modes. ISIM does not currently include the heating mode (i.e., VAV reheat), so this more sophisticated scheme is not implemented; the VAV mode is always cooling. The airflow rate is calculated from a PI loop that compares the zone's temperature, T_z , to the setpoint temperature, $T_{z,sp}$. If T_z is greater than $T_{z,sp}$, the airflow rate will increase, and vice versa. When reheat is included, the heater will turn on once the airflow rate decreases to the minimum and T_z is still less than $T_{z,sp}$. Once the heater reaches its maximum setting, the airflow rate will increase to the maximum airflow rate for heating mode.

Table 13 shows the PI values used in the controller. EU_{Low} is the maximum ventilation airflow rate from HSIM (V_{min}) or the minimum required airflow for the electric heaters (340 m³/hr).

Table 13. Settings for the PI logic for the VAVs in cooling mode.

Name	VAV
K_p	0.5
K_i	0.0001
SP_{High}	32.2 °C (90 °F)
SP_{Low}	10 °C (50 °F)
u_{Bias}	0
A	-1
EU_{High}	1580 m ³ /hr (930 cfm)
EU_{Low}	max(340 m ³ /hr, V_{min})

4.7. AHU static pressure setpoint

The AHU fan speed is controlled to maintain the static pressure in the duct downstream of the AHU at the setpoint. In a typical scenario, the airflow setpoint increases because the zone temperature is above the setpoint. This increase causes the damper in the VAV to open, which decreases the static pressure in the duct. Increasing the fan speed brings the static pressure back to the setpoint. If cooling demand decreases, on the other hand, the airflow setpoint decreases, the VAV damper closes, the static pressure increases and the AHU fan speed decreases to bring the static pressure down to the setpoint.

The static pressure setpoint can be constant, but better performance is achieved by resetting the setpoint. In the IBAL, pressure reset is based on the damper positions. The goal is to set the pressure to a value that keeps the two VAV dampers associated with an AHU nearly fully open. However, getting the damper position correct in ISIM is difficult, so a neural network (NN) model approximates the current static pressure reset approach. The inputs to the NN model are the current static pressure setpoint, airflow rates, and the difference between the zone temperature and zone setpoint, and the output is the new static pressure setpoint. The inputs

are the features that impact the VAV damper position, so they serve as a proxy for that information. These features are also available from either HSIM or another controller in ISIM.

4.8. Candidates for advanced control approaches

The controllers in the system are a mix of local and supervisory controllers. Local controllers are those that directly calculate the position of an actuator, such as the valve controllers. Supervisory controllers determine setpoints, such as the static pressure setpoint, or what equipment to use, such as the chiller staging. Supervisory controllers are good candidates for advanced control approaches and are the target of future work. The current versions are the baseline to which other control approaches will be compared. The control decisions that will be targeted in future work include some described above and some that have not yet been implemented in ISIM:

- chiller staging,
- TES operation,
- chilled water temperature setpoint,
- airflow setpoint,
- static pressure setpoint,
- secondary loop operation,
- secondary loop differential pressure setpoint,
- reheat setpoint, and
- supply air temperature setpoint.

The last three items on this list have not been discussed in detail. The logic behind the secondary loop differential pressure (DP) setpoint is like the static pressure setpoint. If the cooling coil valves are more closed, the DP increases, so the SL pump speed decreases to reach the DP setpoint. If the cooling coil valves are more open, the DP decreases and the SL pump speed increases to bring the DP back up to setpoint. In HSIM, the DP setpoint is a constant of 551.6 Pa, but resetting this setpoint can lead to more efficient operation. Reheat is not currently implemented in ISIM, but several methods exist to determine the reheat temperature setpoint. The supply air temperature (SAT) setpoint is currently set to a constant value of 12.8 °C. This value can also be reset to improve operations.

5. System-level Validation

It is difficult to evaluate the performance of the entire system model of IBASIM using the same approach applied to the component models because there are over 100 temperatures, pressures, flow rates, and powers to consider. At this point, it is important to remember the intended future use of IBASIM: given a control algorithm, IBASIM will be used to determine if that algorithm should be deployed in the IBAL using the actual hardware. This usage does not require that IBASIM is highly accurate; it requires that it captures trends correctly and identifies potential issues in software before moving to hardware. Therefore, the first phase of the analysis examines how the power and cost of different control approaches compare to a baseline. Seven cases from the HILFT project are used for this evaluation. Table 14 lists the cases with key details, including the labels used in the following discussion and figures.

Table 14. Test cases for system-level validation. All test locations are Atlanta, GA, US.

Dataset	Weather	Control Approach	Label
2022_05_05_run2	Extreme summer day	Eff	ex_eff
2022_05_12_run4	Extreme summer day	Shed	ex_shed
2022_09_19_run4	Extreme summer day	Shift	ex_shift
2022_05_10_run4	Typical summer day	Eff	sum_eff
2022_04_01_run1	Typical summer day	Shed	sum_shed
2022_08_31_run3	Typical summer day	Shift	sum_shift
2022_08_09_run1	Typical summer day	Shift with TES	sum_tes

For both the summer and extreme summer days, the efficiency control approach is the baseline against which the other control approaches are compared. Note that the classification of a day as typical or extreme is based on the definitions in an EnergyPlus weather file [30]. The efficiency cases use a rule-based control approach based on ASHRAE Standard 90.1-2004, the key details of which are:

- HVAC system operation (simulation time)
 - Occupied: 6 am – 8 pm, and
 - Unoccupied (setback): 12 am – 6 am, 8 pm – 12 am.
- Zone temperature dual setpoints
 - Occupied: 20 °C – 25.6 °C, and
 - Unoccupied (setback): 12.8 °C – 32.2 °C.
- System-level setpoints
 - AHU Supply Air Temperature: 12.8 °C,
 - AHU Supply Air Static Pressure: trim and respond reset, max = 398.5 Pa,
 - Chilled Water Supply Temperature: reset by outdoor air temperature, [4.4 °C, 8.9 °C],
 - Chilled Water Secondary Loop Differential Pressure Setpoint: 551.6 kPa, and
 - VAV Minimum Air Flow: 34 m³/hr-person (20 cfm/person).

The VAV minimum airflow is set to the maximum of the 34 m³/hr-person rate and the requirements of the IBAL. The IBAL uses electric heaters to generate zone loads and for VAV reheat. These heaters require a minimum airflow rate of approximately 340 m³/hr (200 cfm) to protect the heater elements from burn-out.

The peak period for the Atlanta, GA, location used in these tests is 13:00 to 18:00. The peak cost is 0.16923 \$/kWh, and the off-peak cost is 0.074646 \$/kWh. In the shedding cases, the load is shed during the peak period by increasing the zone cooling setpoint temperature to 26.7 °C. In the shifting cases, the load is shifted by decreasing the setpoint to 23.9 °C for 3 hours before the peak period (i.e., pre-cooling) and then increasing it to 26.7 °C during the peak hours. In the TES case, the TES is charged between 20:00 and 06:00 and then discharged during the peak hours.

5.1. Overall performance

Before examining the results based on how well the simulation captures the trends, Fig. 25 shows boxplots comparing the power predictions to the measured results, as shown in Eq. (32).

$$\mathit{error}_i = \frac{(y_{sim,i} - y_{data,i})}{y_{data,i}} * 100 \% \quad (32)$$

In these and other figures, *pow_tot* is the total system power. The hydronic system power, *pow_hyd*, is the sum of the chiller powers (*ch1_power* and *ch2_power*) and the pump powers (*pump1_power*, *pump2_power*, *pump3_power*). The air system power, *pow_air*, is the sum of the AHU fan powers (*ahu1_fan_power* and *ahu2_fan_power*). The y-axes have been scaled based on the whiskers, so some outliers are cut off. Table 15 shows the median values and the number of outliers for each case as a percentage of the total number of data points. The total system power is generally estimated within 10 %. The *ex_shift* case shows a large underestimation of the power in the air system; the reasons are discussed below. In general, the combined power of the two fans accounts for less than 20 % of the total power, while the chillers generally account for more than 60 % of the power consumption. However, the fan power can become more significant under more advanced control schemes, such as when the TES is in use, so it is still important to have a good model.

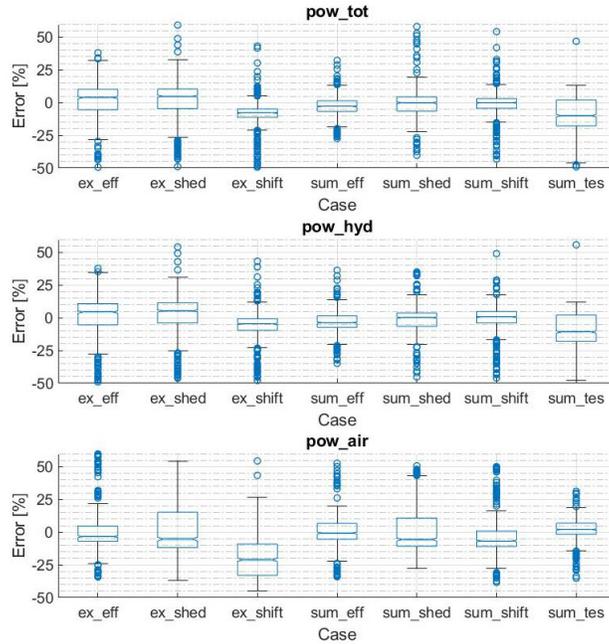


Figure 25 Boxplots of the percent error in the power between the simulation and the data. The y-axes cut off some of the outliers.

Table 15. Number of outliers as a percentage of the total data points and the median percent error.

Case	Outliers [%]			Median [%]		
	pow_tot	pow_hyd	pow_air	pow_tot	pow_hyd	pow_air
ex_eff	3.1	4.1	9.4	3.9	4.4	-3.3
ex_shed	5.1	5.2	5.0	4.9	5.3	-5.3
ex_shift	7.3	6.2	0.2	-7.8	-4.7	-21.1
sum_eff	4.4	4.0	5.9	-2.8	-3.7	-0.7
sum_shed	3.4	4.5	18.3	-0.1	0.1	-5.6
sum_shift	7.5	6.9	7.9	-0.1	0.8	-6.7
sum_tes	1.3	1.8	2.3	-10.0	-10.9	2.1

Figure 26 shows the trend of power usage. The *data* results are the measured values from the HILFT tests, and the *sim* results are the results from IBASIM. The powers are normalized by the baseline power of the summer or extreme summer cases, *sum_eff* and *ex_eff*, respectively.

Figure 27 shows the power usage of the key equipment relative to a baseline. For the components in the hydronic system, the baseline power is *pow_hyd* for either *sum_eff* or *ex_eff*. For the components in the air system, the baseline power is *pow_air*, and for the total power, it is *pow_tot*. The relative costs based on the power consumption are shown in Figure 28.

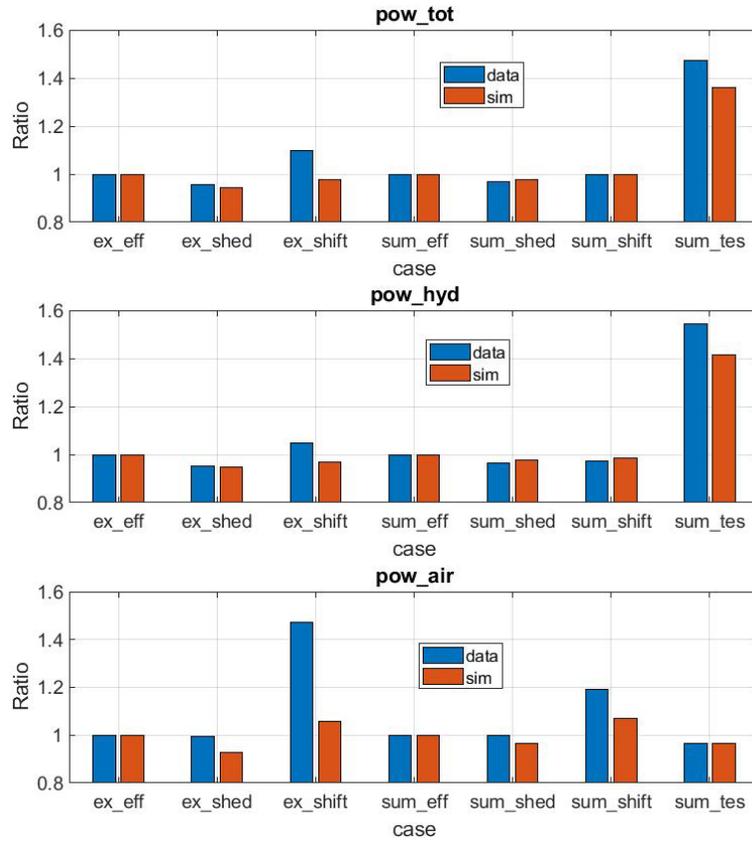


Figure 26 Normalized total power consumption.

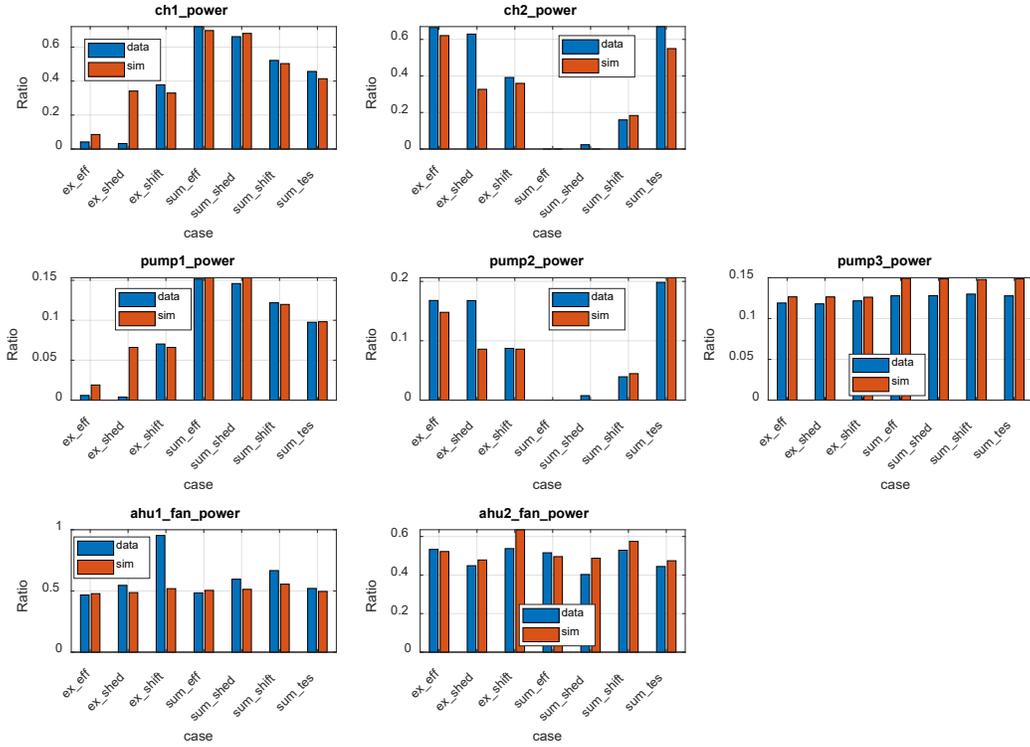


Figure 27 Normalized power for individual components.

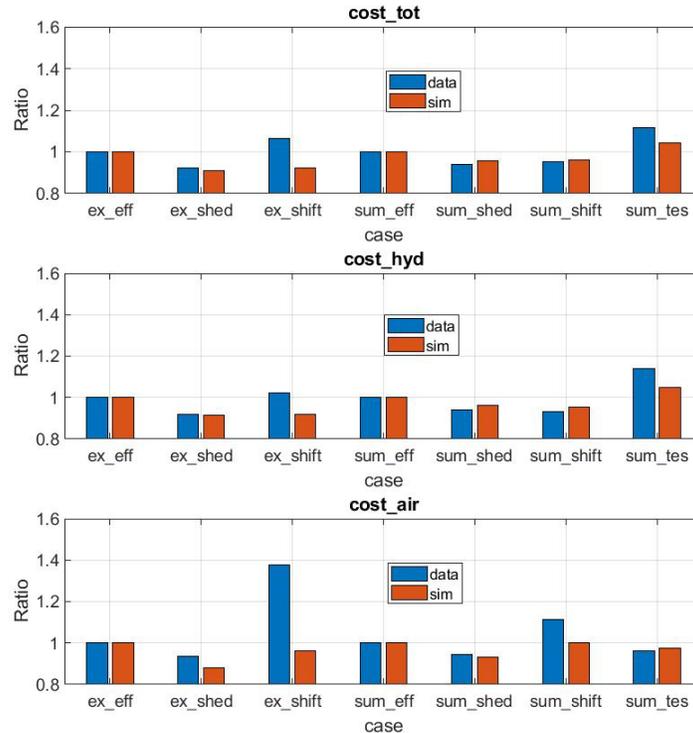


Figure 28 Normalized cost.

The goal of IBASIM is for the ratios to exhibit the same trend as the ratios from HILFT. For example, based on the data and the simulation, the *ex_shed* and *sum_shed* cases use the least total power and have the least total cost, so the shed approach would be towards the top of the list for testing in the IBAL. IBASIM generally shows the same trends, but some results deserve additional scrutiny.

For the shifting cases, particularly the *ex_shift* case, the measured AHU1 fan power consumption (*ahu1_fan_power*) is greater than predicted by the simulation. This underestimate also results in an underestimate of the cost for *ex_shift*. The conclusion from the IBASIM result would, therefore, be that the shifting approach performs better than it did. The cause of the underestimation in fan power is an underestimation in the airflow rate in the model. The airflow rate setpoint is determined from the difference between the zone temperature and the zone temperature setpoint. If the temperature is above the setpoint, the airflow will increase to bring the temperature towards the setpoint, and vice versa. The zone setpoint is similar between the data and the simulation, but the zone temperature is lower in the simulation. That difference is large enough to result in a lower overall airflow rate during the pre-cooling period associated with load shifting. Figure 29 shows the underestimation of airflow rate and zone temperature for Zone3 and Zone4, which are served by AHU1. Once the difference between the temperature and setpoint drops near and below zero, as shown in the middle row of the figures, the airflow rate decreases, leading to lower fan power consumption. This result is discussed further in Sec. 5.2.4.

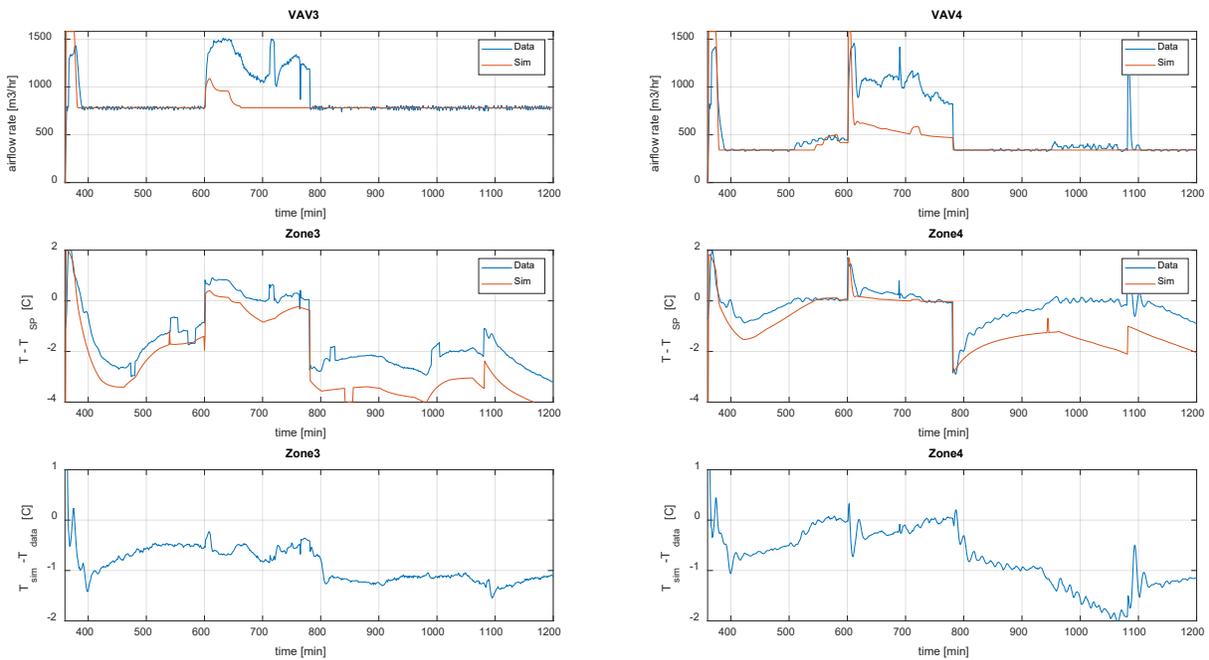


Figure 29 Airflow and temperature data related to the underprediction of fan power during the pre-cooling period for the *ex_shift* case.

The temperature of the air coming off the AHU is lower in the simulation than in the lab, perhaps because the CC model overestimates the effectiveness of the CC. Future work will look for better CC models, possibly using an approach where UA is a function of flow rate instead of a constant value.

The power consumption of the chillers generally shows the right trends, but there are a couple of cases where there is a significant difference. These discrepancies are due to the inconsistency of the chiller staging approach between IBASIM and HILFT, as discussed in Sec. 4.3. This issue is also why the Pump1 and Pump2 power consumption are inconsistent.

5.2. Component performance

This section presents the performance metrics for some components used in the system-level model. When the model performs worse in the system-level validation than in the component-level validation, it can be challenging to determine the causes because all the models are integrated. Therefore, the boundary conditions applied to the component model may be the issue rather than the performance of the component model itself.

5.2.1. Chiller models

Figures 30 and 31 show the system-level metrics for Chiller1 and Chiller2, respectively. These metrics are calculated considering only timesteps when the chiller is on in the HILFT and IBASIM results. The metrics for *ex_eff* and *ex_shed* are not shown because Chiller1 operates for only a short time for those cases. These system-level results are consistent with the component-level validation in Sec. 3.2.1.

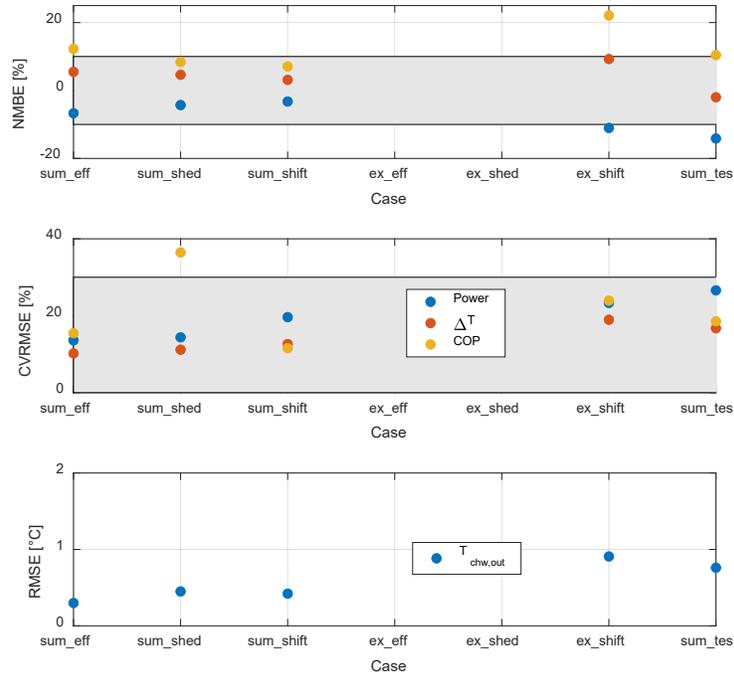


Figure 30 Chiller1 performance for seven cases under system-level validation.

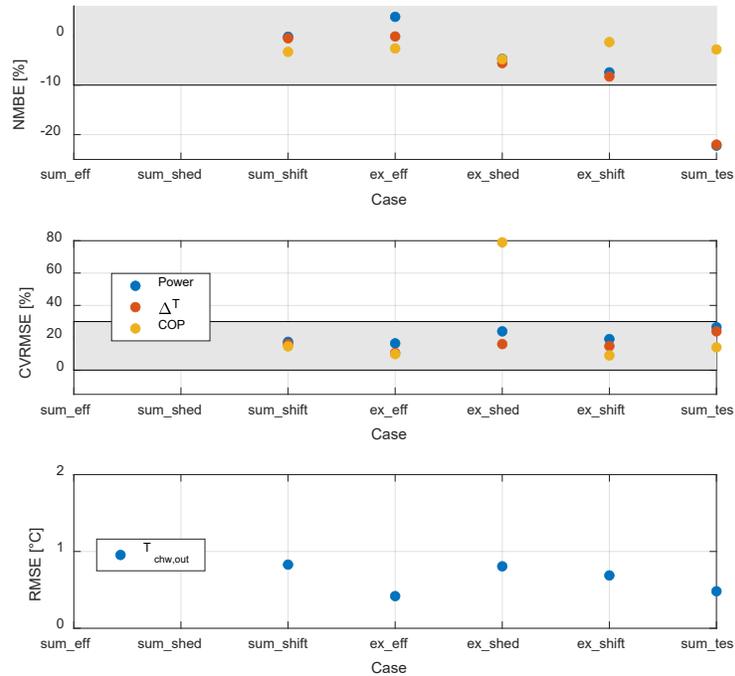


Figure 31 Chiller2 performance for seven cases under system-level validation.

5.2.2. Hydronic flow

Figure 32 shows the results for the hydronic flow model when it is used in the system-level validation. Each flow rate is evaluated only when both IBASIM and HILFT show flow. Recall from Sec. 3.2.2 that the flow rates in the PL are hardcoded, so, as expected, the metrics for Chiller1, Chiller2, and the TES meet the requirements of G14. In the SL, the only value that is well modeled is the flow rate in AHU1. Since the SL flow rate is based on the sum of AHU1 and AHU2, it is wrong primarily because the AHU2 flow is wrong.

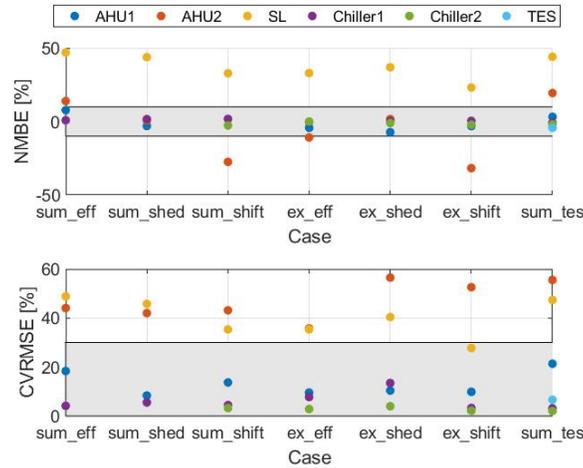


Figure 32 Hydronic flow results when used in the system-level validation.

The flow calculation in the AHUs is based on the pressure drop through the valves. The IBAL does not currently have dedicated measurements of the pressure drop through individual valves, so the parameter values are based on a pressure drop measurement across large portions of the piping in the system at different valve positions. Better measurements are required to obtain better parameters. In addition, the pressure drop calculation requires the valve position as an input, so the error in the valve model also contributes to the error in the flow rate model. Figure 33 shows the metrics for the models of the cooling coil valves in AHU1 (V12), AHU2 (V13), and the TES valve (V8). The models for V12 and V8 generally perform well, but V13 performs poorly. It is unclear why the performance of V13 is so much worse than that of V12 since they use the same control logic. One possibility is that the load is higher and more variable in AHU2, and the relatively simple model does not sufficiently capture this large range in the dynamics. The key metric here, however, is the heat transfer rate of the cooling coil. The performance of that metric is discussed in Sec. 5.2.5.

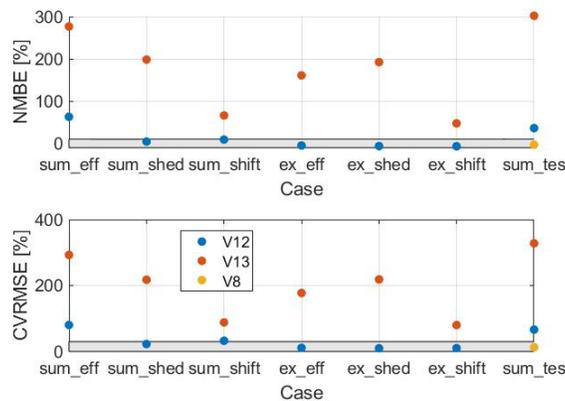


Figure 33 Valve model results when used in system-level validation.

5.2.3. Pump power

Figure 34 shows the pump power metrics when used in the system-level simulation. The results are similar to those of the component-level validation.

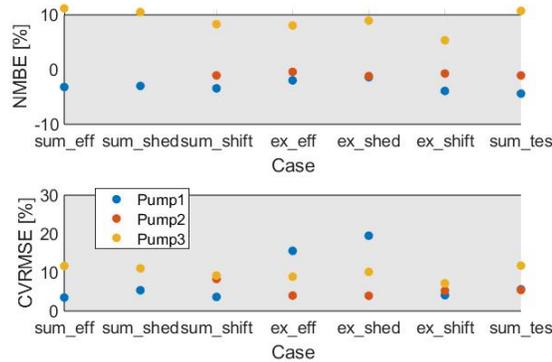


Figure 34 Pump power results when used in system-level validation.

5.2.4. AHU fan power

Figure 35 shows the AHU fan power metrics from the system-level validation. In the component-level validation, the models passed met the G14 thresholds, but the performance at the system level is worse. The cause of this issue is the error in the inputs to the model. Figures 36 and 37 show the metrics for the inputs to the AHU1 and AHU2 models, respectively. The primary issue is predicting the difference between the zone temperature and the zone setpoint. This result was discussed previously (see Fig. 29).

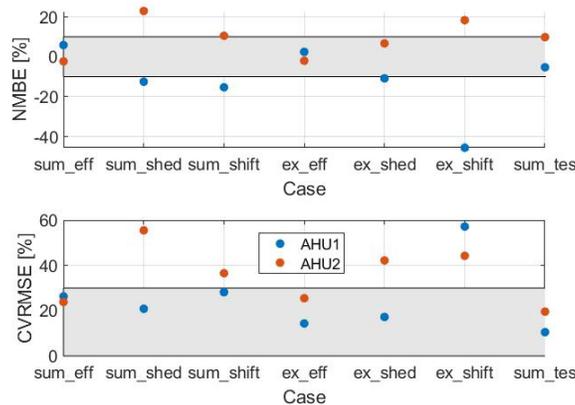


Figure 35 AHU fan power results when used in system-level validation.

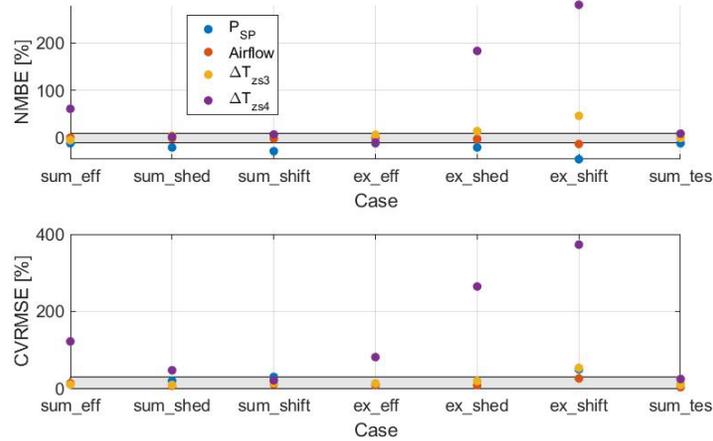


Figure 36 Performance metrics for the inputs to the AHU1 fan power model.

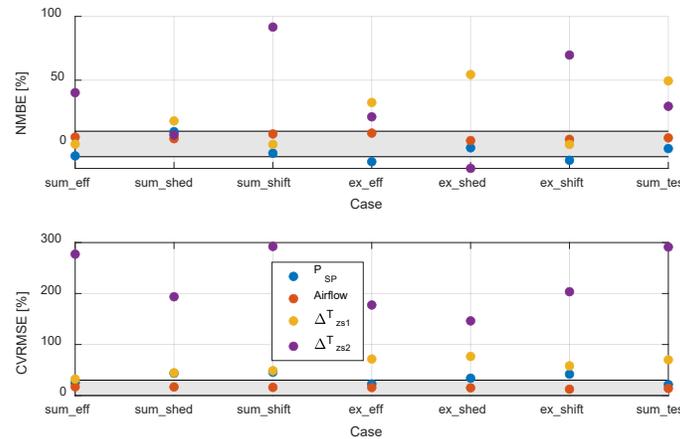


Figure 37 Performance metrics for the inputs to the AHU2 fan power model.

5.2.5. AHU cooling coil

Figure 38 shows the system-level performance of the cooling coil models. The legend labels are:

- $AHU\#_{air}$ = difference between the air temperatures downstream and upstream of the AHU,
- $AHU\#_{pg}$ = difference between the temperature of the propylene glycol leaving and entering the cooling coil,
- $AHU\#_q$ = heat transfer rate on the liquid side of the cooling coil,
- $AHU\#_{air,out}$ = temperature of the air downstream of the AHU,
- $AHU\#_{air,in}$ = temperature of the air upstream of the AHU, and
- $AHU\#_{pg}$ = temperature of the propylene glycol leaving the cooling coil.

The temperature entering the cooling coil is not calculated in this subsystem and is therefore not evaluated here. The air temperature differences meet the G14 acceptance criteria except

for the ex_eff case, while the propylene glycol temperature differences show more variation in accuracy. Looking at the RMSE values of the temperatures, the errors are largest for the inlet air temperature and the leaving propylene glycol temperature. The larger RMSE values for these temperatures are not surprising because the system is highly coupled, and errors will build upon each other. These results are consistent with the findings from Sec. 3.2.4.

The errors in the heat transfer rates consistently exceed the acceptance criteria. The issue is that the error is a combination of the errors in the flow rate, the temperature of the fluid leaving the cooling coil, and the temperature of the fluid entering the cooling coil, so the combined error can be significant. Future work will consider a more direct heat transfer model that relies less on the accuracy of other models.

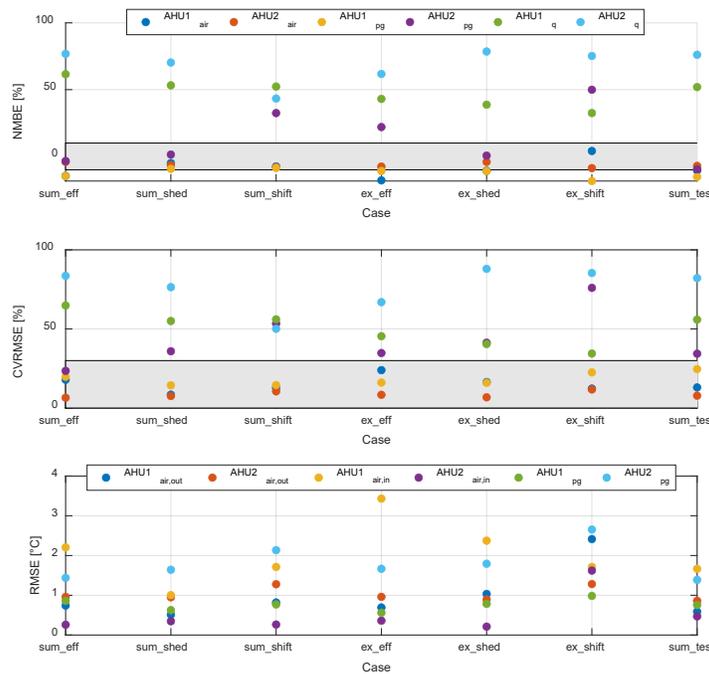


Figure 38 AHU cooling coil results when used in system-level validation.

5.2.6. Control decisions

This section presents the metrics for some supervisory-level control decisions discussed in Sec. 4. Figure 39 shows the metrics for the on- and off-control decisions for the SL, each chiller, and the TES. As previously mentioned, the chiller staging controller does not perfectly match the lab because the chiller models do not fully capture the transient behavior of the actual equipment, so the staging decisions in IBASIM are made differently from those in the actual laboratory. The controllers for the TES and SL, on the other hand, perform very well.

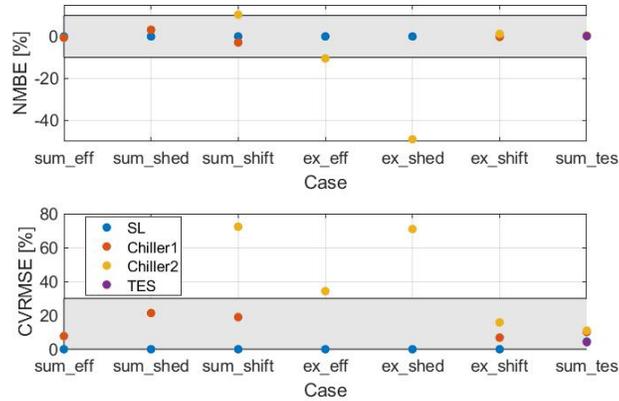


Figure 39 Performance of the on/off control decisions.

Figure 40 shows the metrics for the VAV airflow controllers. In general, these controllers meet the acceptance criteria. However, the CVRMSE values are on the higher end of the range, and since the airflows feed into the heat transfer and power calculations, it could be worthwhile to develop more accurate controllers. However, the first step would be to improve the cooling coil models since the zone temperature feeds into the VAV airflow control calculation.

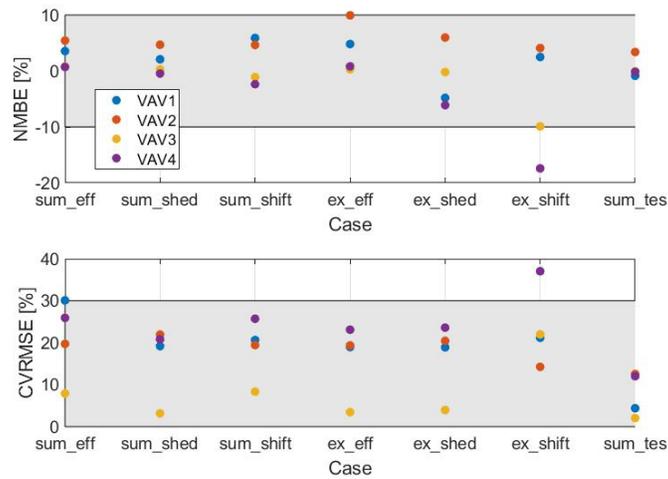


Figure 40 VAV airflow rate results when used in system-level validation.

6. Conclusion

This report has presented the details of the IBASIM architecture and the calibration and validation of the model (see Figure 1). Although some of the individual aspects of the system are not modeled within the accuracy specified by G14, the overall assessment of the system-level performance based on power and cost would lead to the correct evaluation of the potential of the control strategies evaluated in this study. The conclusion, therefore, is that this model can be used to develop and assess the potential of different control approaches, but it should be updated and reassessed as new data becomes available. The models that should be further developed as time and resources allow include those listed in Table 16. The priority is based primarily on the impact of that model on the overall evaluation of the controller performance. One important observation is that there are control approaches, such as those in Guideline 36, that can rely on valve and damper positions to determine and adjust some setpoints. So, if an approach requires that information, those models should also be updated to capture the actual behavior of those devices better.

Table 16. Models for further development.

Model	Limitation	Possible approaches	Priority
Chiller	Does not fully capture transient and dynamic behavior	Re-optimize parameter values; ML approach	Low
Primary loop flow rates	Hard-coded constant values for the two chillers	Improved data for pressure measurements; ML approach	High
Secondary loop flow rates	Poor accuracy	Improved data for pressure measurements; ML approach	High
Pump power	Calibrated and validated over a limited operating range	Re-optimize parameters using a larger dataset	Low
AHU fan power	Poor accuracy	An approach that uses fewer input variables; an approach similar to the pump power model	High
AHU cooling coil	Poor accuracy	Variable UA; ML approach	Medium

In addition, although IBASIM does include models for HX1 and VAV reheat, they have not been validated against larger system-level datasets. The one component that is completely missing from IBASIM is the water-side economizer, which needs to be added to investigate that mode of operation. There is also the potential to add simulated systems that enable more extensive research into control options. In the real world, the condensing water for the chillers is supplied by a cooling tower, so it would be helpful to add a simulation of a cooling tower to the system to better understand how to optimize the entire cooling plant. Another possibility is to add simulated solar panels to see how onsite electricity generation impacts control decisions.

References

- [1] EIA (2023) Energy use in commercial buildings. <https://www.eia.gov/energyexplained/use-of-energy/commercial-buildings.php>.
- [2] Kanagala KR, Pertzborn A (2023) Comparison of Ice-on-Coil Thermal Energy Storage Models. <https://doi.org/10.6028/NIST.TN.2265>
- [3] Salimian Rizi B (2023) Data-Driven Modeling for Advancing Near-Optimal Control of Water-Cooled Chillers. PhD. (IIT, Chicago, IL).
- [4] ASHRAE (2024) Titles, Purposes, and Scopes: SPC 223P .
- [5] Pertzborn AJ (2024) IBA-Project: Code repository. Available at <https://github.com/usnistgov/IBA-Project>
- [6] Pertzborn AJ (2016) NIST Technical Note 1933: Intelligent Building Agents Laboratory : Hydronic System Design. <https://doi.org/http://dx.doi.org/10.6028/NIST.TN.1933>
- [7] Pertzborn AJ, Veronica DA (2018) NIST Technical Note 2025: Intelligent Building Agents Laboratory: Air System Design. <https://doi.org/https://doi.org/10.6028/NIST.TN.2025>
- [8] NIST (2021) Intelligent Building Agents Project. <https://www.nist.gov/el/energy-and-environment-division-73200/intelligent-buildings-agents-project>.
- [9] Kopach A, Nellis G, Bradley D, Reindl D, Pertzborn A (2021) Modeling of the NIST Intelligent Building Agents Laboratory. Available at <https://engineering.purdue.edu/Herrick/Events/orderlit.html>
- [10] Abdulmajeid L (2023) MODELING AND CONTROL OF THE INTELLIGENT BUILDING AGENTS LABORATORY. MS (University of Wisconsin-Madison, Madison, WI).
- [11] Kopach A (2021) Modelling the Air-Side of the Intelligent Building Agents Laboratory. MS (University of Wisconsin-Madison, Madison, WI).
- [12] Pradhan O, Pertzborn A, Zhang L, Wen J (2020) Development and validation of a simulation testbed for the Intelligent Building Agents Laboratory (IBAL) using TRNSYS. *ASHRAE Transactions*, Vol. 126.
- [13] Chen Z, Wen J, Bushby ST, Member A, Member A, Fellow A, Lo ALJ, O'Neill Z, Vance Payne W, Pertzborn A, Calfa C, Fu Y, Grajewski G, Li Y, Yang Z (2022) Development of a Hardware-in-the-loop Testbed for Laboratory Performance Verification of Flexible Building Equipment in Typical Commercial Buildings. (ASHRAE 2022 Annual Conference, Toronto, Ontario, CA).
- [14] Fu Y, O'Neill Z, Wen J, Pertzborn A, Bushby ST (2022) Utilizing commercial heating, ventilating, and air conditioning systems to provide grid services: A review. *Applied Energy* 307. <https://doi.org/10.1016/j.apenergy.2021.118133>
- [15] ASHRAE (2004) 90.1-2004: Energy Standard for Buildings Except Low-Rise Residential Buildings (2004). (Atlanta, GA).
- [16] U.S. Department of Energy (2023) Commercial Prototype Building Models. <https://www.energycodes.gov/prototype-building-models>.
- [17] Crawley DB, Lawrie LK, Winkelmann FC, Buhl WF, Huang YJ, Pedersen CO, Strand RK, Liesen RJ, Fisher DE, Witte MJ, Glazer J (2001) EnergyPlus: creating a new-generation building energy simulation program. 319–331.

- [18] Klein SA, Beckman WA, Mitchell JW, Duffie JA, Duffie NA, Freeman TL, Mitchell JC, Braun JE, Evans BL, Kummer JP, Urban RE, Fiksel A, Thornton JW, Blair NJ, Williams PM, Bradley DE, McDowell TP, Kummert M, Arias DA, Duffy MJ, Weiss AM (2017) TRNSYS18: A Transient System Simulation Program. (Madison, WI, USA). Available at <http://www.trnsys.com/>
- [19] The MathWorks Inc. (2021) *MATLAB* (Natick, Massachusetts, United States), R2021a. Available at <https://www.mathworks.com>
- [20] The MathWorks Inc. (2021) *Simulink*, R2021a. Available at <https://www.mathworks.com>
- [21] Pertzborn A (2022) IBAL Database. <https://ibal.nist.gov/>.
- [22] Lawrence Berkeley National Laboratory (2020) FMU Export of EnergyPlus. <https://simulationresearch.lbl.gov/fmu/EnergyPlus/export/>.
- [23] Langevin J, Wen J, Gurian PL (2015) Simulating the human-building interaction: Development and validation of an agent-based model of office occupant behaviors. *Building and Environment* 88:27–45. <https://doi.org/10.1016/j.buildenv.2014.11.037>
- [24] Langevin J, Gurian PL, Wen J (2015) Tracking the human-building interaction: A longitudinal field study of occupant behavior in air-conditioned offices. *Journal of Environmental Psychology* 42:94–115. <https://doi.org/10.1016/j.jenvp.2015.01.007>
- [25] ASHRAE (2023) ASHRAE Guideline 14-2023: Measurement of energy, demand and water savings. (Peachtree Corners, GA).
- [26] TESS (2018) *TESS COMPONENT LIBRARIES General Descriptions*. Available at <https://www.trnsys.com/tess-libraries/>
- [27] Pertzborn AJ, Veronica DA (2022) Baseline Control Systems in the Intelligent Building Agents Laboratory. (Gaithersburg, MD). <https://doi.org/10.6028/NIST.TN.2178>
- [28] Taylor ST, Stein J, Paliaga G, Cheng H (2012) Dual Maximum ASHRAE Journal. *ASHRAE Journal*.
- [29] Hydeman M, Eubanks B (2014) ASHRAE Research Project Report 1455-RP: Advanced Control Sequences for HVAC Systems-Phase I Air Distribution and Terminal Systems. Available at www.ashrae.org.
- [30] U.S. DOE (2022) EnergyPlus Version 22.1.0 Documentation: Auxiliary Programs. Available at https://energyplus.net/assets/nrel_custom/pdfs/pdfs_v22.1.0/AuxiliaryPrograms.pdf

Appendix A. IBAL Data Dictionary

The measurements referenced throughout this report are defined in Table 17.

Table 17. Definitions of measurements.

Name	Description
ahu1_cc_out_rtd	Temperature of the propylene glycol leaving the cooling coil in AHU1
ahu1_f_cc	Flow rate of PG through the cooling coil in AHU1
ahu1_f_ra	Return airflow rate in AHU1
ahu1_f_sa	Supply airflow rate in AHU1
ahu1_fan_power	Power consumption of the AHU1 supply fan
ahu1_in_rtd	Temperature at the inlet of AHU1, after the return and supply air are mixed
ahu1_out_rtd	Temperature downstream of AHU1
ahu1_p_down	Duct static pressure downstream of AHU1
ahu1_p_up	Static pressure upstream of the supply fan in AHU1
ahu1_rh_down	Relative humidity downstream of AHU1
ahu1_rh_ra	Relative humidity of the return air in AHU1
ahu1_rh_sa	Relative humidity of the supply air in AHU1
ahu1_rh_up	Relative humidity upstream of the supply fan in AHU1, after the return and supply air are mixed
ahu1_t_ra	Temperature of the return air in AHU1
ahu1_t_sa	Temperature of the supply air in AHU1
ahu1_vfd	Speed of the AHU1 supply fan
ahu2_cc_out_rtd	Temperature of the propylene glycol leaving the cooling coil in AHU2
ahu2_f_cc	Flow rate of PG through the cooling coil in AHU2
ahu2_f_ra	Return airflow rate in AHU2
ahu2_f_sa	Supply airflow rate in AHU2
ahu2_fan_power	Power consumption of the AHU2 supply fan
ahu2_in_rtd	Temperature at the inlet of AHU2, after the return and supply air are mixed
ahu2_out_rtd	Temperature downstream of AHU2

Name	Description
ahu2_p_down	Duct static pressure downstream of AHU2
ahu2_p_up	Static pressure upstream of the supply fan in AHU2
ahu2_rh_down	Relative humidity downstream of AHU2
ahu2_rh_ra	Relative humidity of the return air in AHU2
ahu2_rh_up	Relative humidity upstream of the supply fan in AHU2, after the return and supply air are mixed
ahu2_t_ra	Temperature of the return air in AHU2
ahu2_t_sa	Temperature of the supply air in AHU2
ahu2_vfd	Speed of the AHU2 supply fan
ch1_c_out_rtd	Temperature of the water leaving the condenser in Chiller1
ch1_e_in_rtd	Temperature of the PG returning to both Chiller1 and Chiller2
ch1_e_out_rtd	Temperature of the PG leaving the evaporator in Chiller1
ch1_f_c	Flow rate of the water through the condenser in Chiller1
ch1_f_e	Flow rate of the PG through the evaporator in Chiller1
ch1_power	Power consumption of Chiller1
ch2_c_out_rtd	Temperature of the water leaving the condenser in Chiller2
ch2_e_out_rtd	Temperature of the PG leaving the evaporator in Chiller2
ch2_f_c	Flow rate of the water through the condenser in Chiller2
ch2_f_e	Flow rate of the PG through the evaporator in Chiller2
ch2_power	Power consumption of Chiller2
hx1_f_pg	Flow rate of the PG in HeatExchanger1
hx1_pg_rtd	Temperature of the PG at the outlet of HeatExchanger1
pl_out_rtd	Temperature of the PG leaving the primary loop
pump1_power	Power consumption of Pump1
pump2_power	Power consumption of Pump2
pump3_dp	Differential pressure across the secondary loop
pump3_out_rtd	Temperature downstream of Pump3
pump3_power	Power consumption of Pump3
sl_f	Flow rate in the secondary loop

Name	Description
sl_out_rtd	Temperature at the outlet of the secondary loop
ts_f	Flow rate through the thermal storage tank
ts_in_rtd	Temperature of the PG entering the thermal storage tank
ts_out_rtd	Temperature of the PG leaving the thermal storage tank
vav1_f	Airflow rate through VAV1
vav1_heat_power	Power consumption of the electric reheat in VAV1
vav1_out_rtd	Temperature of the air downstream of VAV1
vav1_t_in	Temperature of the air upstream of VAV1
vav2_f	Airflow rate through VAV2
vav2_heat_power	Power consumption of the electric reheat in VAV2
vav2_out_rtd	Temperature of the air downstream of VAV2
vav2_t_in	Temperature of the air upstream of VAV2
vav3_f	Airflow rate through VAV3
vav3_heat_power	Power consumption of the electric reheat in VAV3
vav3_out_rtd	Temperature of the air downstream of VAV3
vav3_t_in	Temperature of the air upstream of VAV3
vav4_f	Airflow rate through VAV4
vav4_heat_power	Power consumption of the electric reheat in VAV4
vav4_out_rtd	Temperature of the air downstream of VAV4
vav4_t_in	Temperature of the air upstream of VAV4
zs1_out_rtd	Temperature of the air downstream of ZS1 (zone temperature)
zs1_rh	Relative humidity of the air downstream of ZS1 (zone relative humidity)
zs2_out_rtd	Temperature of the air downstream of ZS2 (zone temperature)
zs2_rh	Relative humidity of the air downstream of ZS2 (zone relative humidity)
zs3_out_rtd	Temperature of the air downstream of ZS3 (zone temperature)
zs3_rh	Relative humidity of the air downstream of ZS3 (zone relative humidity)
zs4_out_rtd	Temperature of the air downstream of ZS4 (zone temperature)
zs4_rh	Relative humidity of the air downstream of ZS4 (zone relative humidity)

Appendix B. List of Symbols, Abbreviations, and Acronyms

dp Subscript that indicates a differential pressure measurement	GA Genetic algorithm
f Subscript that indicates a flow measurement	G14 ASHRAE Guideline 14
p Subscript that indicates a pressure measurement	HILFT Hardware-in-the-loop load flexibility testing
rh Subscript that indicates a relative or absolute humidity measurement	HSIM HILFT portion of the full simulation platform, IBASIM. HILFT Simulation.
rtd Subscript that indicates a temperature measured by a resistance temperature detector	HVAC Heating, ventilation, and air conditioning
sim Simulation or model results	HX Heat exchanger
AHU Air handling unit	IBAL Intelligent Building Agents Laboratory
CC Cooling coil	IBASIM Full simulation platform
COP Coefficient of performance	ISIM Components of the full simulation platform, IBASIM, modified or added by NIST. IBAL Simulation.
CVRMSE Coefficient of variation of root mean squared error	J Objective function
D Damper	MBE Mean bias error
DOE Department of Energy	ML Machine learning
DOP Definition	NMBE Normalized mean bias error
DP Differential pressure	OAU Outdoor air unit; used for weather emulation
EOA Emulated outdoor air	OBM Occupant behavior model
FMU Functional mockup unit	PG Propylene glycol; 30 % concentration
	PI Proportional integral

PL

Primary loop

PSO

Particle swarm optimization

RMSE

Root mean squared error

SA

Simulated annealing

SAT

Supply air temperature

SL

Secondary loop

TES

Thermal energy storage

TRNSYS

TraNsient SYstem Simulation software

TSIM

TRNSYS simulation component of the full simulation platform, IBASIM

UA

Overall heat transfer coefficient times area

V

Valve

VAV

Variable air volume

VBM

Virtual building model

ZS

Zone emulator (aka, simulator); used for building load emulation