



Hybrid Approach for Digital Twins in the Built Environment

Yu-Wen Lin
yuwen.lin@berkeley.edu
University of California, Berkeley
Berkeley, CA, USA

Tsz Ling Elaine Tang
tszling.tang@siemens.com
Siemens Technology
Princeton, NJ, USA

Costas J. Spanos
spanos@berkeley.edu
University of California, Berkeley
Berkeley, CA, USA

ABSTRACT

In recent years, several countries have created policies to enforce Zero Energy Building (ZEB) standards for new building construction. Achieving this for new buildings is feasible, but it may be difficult to transform existing buildings to ZEBs. Digital twins provide a promising approach to monitor existing buildings and further increase their energy efficiency. A digital twin (DT) is a virtual representation of a physical entity. It has several applications in product design, product cycle, and fault detection. This paper presents a hybrid approach that combines physics-based and machine learning methods to create a DT for the built environment. A case study for a digital twin of a single room is also discussed. The initial comparison of cooling energy between the physical testbed and the DT model shows promising results for future development. The limitations, challenges, and future development of the approach are also addressed in the paper.

CCS CONCEPTS

• **Computing methodologies** → **Modeling and simulation; Model development and analysis.**

KEYWORDS

digital twin, building modeling, hybrid models

ACM Reference Format:

Yu-Wen Lin, Tsz Ling Elaine Tang, and Costas J. Spanos. 2021. Hybrid Approach for Digital Twins in the Built Environment. In *The Twelfth ACM International Conference on Future Energy Systems (e-Energy '21)*, June 28–July 2, 2021, Virtual Event, Italy. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3447555.3466585>

1 INTRODUCTION

Buildings account for more than 40% of energy consumption around the world [2]. Several research efforts have been done to increase building energy efficiency in the area of Heating, Ventilation and Air-Conditioned (HVAC) systems control [28], lighting [12], and fault detection [16]. Besides reducing energy demand of buildings, in recent years, countries have been moving towards constructing Zero Energy Buildings (ZEB) to further reduce energy consumption through fossil fuels. ZEB is a type of building that produces green energy that meets the energy demand requires for the building [11]. New policies have been introduced in different countries to

help realize ZEB. The recent European Union (EU) policy, Energy Performance of Buildings Directives (EPBD), requires all new buildings from 2021 be nearly ZEB [14]. California plans to have all new commercial construction be Zero Energy Buildings (ZEB) by 2030 [4].

However, for existing buildings, other measures need to be taken. The term *Digital Twin* (DT) emerged in 2012, where it was defined as a virtual space that fully describes the corresponding physical system [9]. In other words, information available in the physical space can be obtained from its virtual DT. Their data is in-sync with each other. The *twin* systems originated from NASA, where they kept a *physical* twin on the ground to help troubleshoot any issues that happen in the space [19]. *Digital* twin was first implemented in GE Research [15] in jet engine. They connect sensors information between physical and virtual systems to help monitor system faults and maintenance needs. DT technology opens up greater opportunities to monitor energy consumption of existing buildings. Furthermore, it can help existing buildings transition to ZEBs.

DT applications lie broadly in product design, production, and prognostics, and health management [22]. DT facilitates manufacturers by enabling them to make more accurate predictions, make rational decisions, and reduce man-kind errors by providing well-rounded information. Currently, most DT research has been focusing on smart manufacturing, where they use DT to improve product planning and fault detection. Tao et. al. studied data fusion techniques of a DT shop-floor between the physical and virtual space [23]. Cai et. al. developed a DT of vertical milling machine by integrating manufacturing and sensor data [3].

Limited research studies have been made for DTs of the built environment. There are research studies that build DTs of energy management in a smart city where they monitor smart meter electricity data of buildings [7]. However, besides electricity consumption, no other information is available for buildings in this study. In addition, most existing research discusses the framework of building a DT, but lacks empirical experiments on the topic.

There are two main approaches for building a DT in built environments: bottom-up and top-down. The Bottom-up approach refers to first-principle modeling where the model captures the dynamics of the physical systems. On the contrary, top-down approach neglects the internal behavior of the systems and creates a black box model of the system via Machine Learning (ML).

There are limited studies for bottom-up approaches in the built environment due to complicated systems and it's usually hardly scalable or generalizable. Aivaliotis et. al. developed physics-based model of an industrial robot hand for predictive maintenance [1]. Prawiranto et. al. developed physics-based DT to monitor solar drying processes [17]. There exists building modeling software such as EnergyPlus [6], which is a physics-based approach software that estimates the energy consumption of a building. It captures



This work is licensed under a Creative Commons Attribution International 4.0 License. *e-Energy '21*, June 28–July 2, 2021, Virtual Event, Italy
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8333-2/21/06.
<https://doi.org/10.1145/3447555.3466585>

Table 1: Comparison between physics-based, machine learning, and hybrid approaches.

	Advantages	Disadvantages
Physics-based	High fidelity High interpretability Extrapolation capacity	Difficult to Construct Computational expensive High uncertain parameters
Machine Learning	Easier to construct Less uncertain parameters	Low interpretability Risk over-fitting Limited extrapolation capacity
Hybrid	Can construct at the proper level of abstractions Reduce the number of uncertain parameters Moderate interpretability	Can be computational expensive

all the physical properties of the system and calculate the energy consumption based on heat balance equation. However, it is not built for DT purpose where the parameters are in-sync with the physical system. Its main purpose is for construction planning and energy consumption prediction. In addition, some of the parameters required in EnergyPlus are often not readily available in the physical system, which results in a large amount of uncertain parameters.

On the other hand, several companies develop DT mainly through data integration and data fusion which use the top-down approach. For example, Johnson Control [5], Enertiv [25], and Willow [24] has developed DT that includes spatial information, static information and real-time sensor data for the purpose of remote and fault monitoring. Siemens developed DT for power grid in Finland that helps improve data utilization and optimize operation [20]. Microsoft has created a platform, Azure Digital Twins, to support modeling any environments with the ability to connect the model to Internet of Things (IoT) devices [21].

While physics-based models have advantages in high fidelity, interpretability, and extrapolation capacity, building a physics-based model requires a deep understanding of a system's structures and operations and is therefore often difficult to construct. Additionally, when constructing a physics-based model, obtaining every required parameter from the physical system can be difficult. This increases the number of uncertainty parameters and can be computationally expensive to solve the systems of equations. In juxtaposition with physics-based models, machine learning models are easier to construct and can be created based on available sensor data. However, machine learning models are rarely physically interpretable, may suffer from over-fitting, and have limited extrapolation capacity. Therefore, we proposed a hybrid approach that utilizes both physics-based modeling and modern predictive machine learning to develop DT. Table 1 summarizes the advantages and disadvantages of the three different approaches.

The integration of physics and machine learning methods is not a new technique [27], but there are limited studies in DT of the built environments using this hybrid approach. As a result, we aim to develop a hybrid DT model that takes advantages of each approach to capture the dynamic behavior of the HVAC system in the building for the purpose of energy efficiency and occupants satisfaction.

The paper is organized as follows. Section 2 describes the approach we use to create hybrid DT models. Section 3 shows an

implementation of the proposed approach for a single-room testbed. Lastly, section 4 and 5 discuss the current challenges and future directions of the research.

2 APPROACH

Our goal is to establish a live connection between the physical and virtual space via sensor data in the built environment for the purpose of energy efficiency and occupants' comfort.

We segment the development of DTs of the dynamic behavior of the HVAC system of a building into three distinct phases as shown in Figure 1. In phase I, we identify the components required to build the model of the physical systems. In phase II, we extract the parameters needed from the physical systems and apply them to the model. We will cycle between Phase I and Phase II until the model can represent the physical system at a proper level of abstraction. Lastly, in phase III, we build the connection between the model and the physical systems for continuous monitoring and updates. The details for each phase are described.

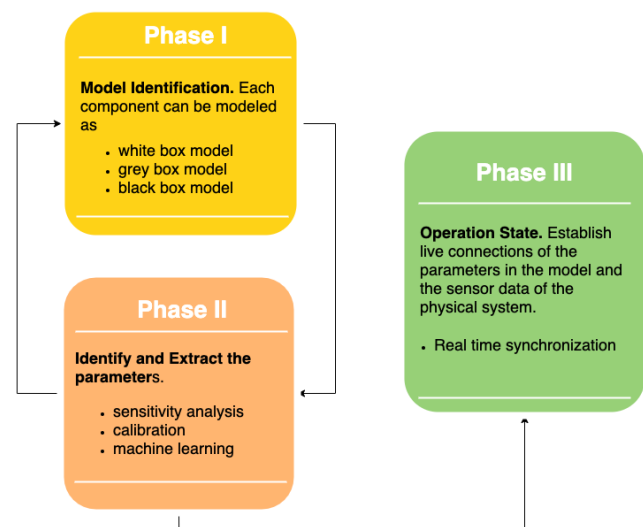


Figure 1: The three phases for developing DT of a built environment with a hybrid approach.

2.1 Phase I: Model Identification

In the first phase, we identify the components that are required to create a complete system. We focus on parameters that are directly available from the sensors in the physical system and pin down other parameters that are essential but not accessible without further effort. We want to break physical components into functional sub-blocks at a proper level. The model is built with OpenModelica [8], which is an open-source software that supports component based modeling and simulates complex dynamic systems. In addition, modelica building library [26] developed by Lawrence Berkeley Lab was used to create the model. The library contains models of components for building and control systems.

Each element in the Modelica model can be modeled as a *black box* (no knowledge of the internal systems), a *grey box* (partial knowledge of the system, i.e. thermal envelope resistance of the wall) or *white box* (all relevant information for the system is available), based on the information we can obtain from the physical system.

This flexible approach would allow us to develop useful models quickly and refine them over time as additional information becomes available. Further, the modularity of this architecture will allow us to readily create and share libraries of parameterized models so that new hybrid models can be quickly synthesized.

2.2 Phase II: Identify and Extract the Parameters

In the second phase, we identify and extract the parameters that are needed for the model from the physical systems. For the black box components, we will perform function approximation using ML, where we only have the inputs and outputs of a particular component and do not know the structure of the internal system. Experimental data can be used to train the model. For example, a neural network with a single or two hidden layers can usually approximate the input-output relationship well. However, when fitting the data into a model, one needs to be careful of over-fitting issues.

For grey box model, we'll use physics-based machine learning to estimate the value needed. For example, it's hard to obtain accurate physical properties (e.g. wall materials and resistance) of the walls in a room. One solution is to create a grey box model using simple thermal circuit equation to estimate the envelope wall resistance of the room.

Next, sensitivity analysis was performed to identify uncertain parameters. This helps reduce the number of uncertain parameters that need to be calibrated. Morris method [13] was used to carry out the sensitivity analysis and it is implemented by utilizing the open source Python Library, SALib [10]. Then, we perform calibration to match the model closely with the physical space.

2.3 Phase III: Operation State

Lastly, we need to establish a connection between the physical entity and the DT model. The modeling parameters should be continuously updated with the gathered sensor data at an experimentally determined frequency. Figure 2 shows the data flow between the physical entity and the DT model. Ideally, when input values such as room set point temperature and supply air temperature are

provided to both physical and DT systems, the resulting impacts to the overall system should be mirrored. The DT model will also receive additional inputs from the physical entity that are not the settings of the system, such as intermediate sensor data, occupancy, and equipment usage. The output values (e.g. cooling energy or electricity consumption) will then be compared for further tuning or monitoring the physical space. This ensures that the DT model is continuously adapting to the change in the physical environment and reduce the model deviation to a minimum. In addition, DT can act as a platform for testing control algorithms and help in decision making in the physical system.

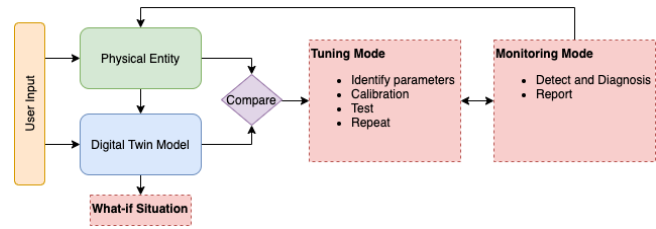


Figure 2: The block diagram showing the data stream between the physical entity and the digital twin model.

3 CASE STUDY

3.1 Physical Testbed

The digital twin of a room in a building was created in a well-instrumented testbed located in Singapore. The indoor environment is shown in Figure 3. The testbed has the capability for high precision control and operations and the indoor environment is well-regulated. The size of the testbed was 25 m² with a height of 2.6m. In contrast with a regular room in a building, the testbed was insulated from the outdoor weather. However, there exists an outdoor air emulator which emulates the outdoor environmental conditions.



Figure 3: The indoor space of the testbed located in Singapore that is used for creating a digital twin.

There are more than 100 different kind of sensors located in various places in the testbed. Table 2 shows a list of available sensors in the testbed. Note that this is not an exhaustive list, the sensors of

different categories are placed in several locations throughout the entire system. We divided the sensors placement into four different categories: room water loop, Air Handling Unit (AHU) water loop, and Outdoor Air Emulator (OAE). The sensor data is automatically uploaded to the PI system¹, a software developed by OSIsoft that connects sensor data to a cloud environment to allow remote access.

Table 2: Available sensor measurements for the testbed.

Category	Parameters
Room Water Loop	Water Flow Rate
	Bypass Valve Opening
	Chilled Water Supply Temperature
	Chilled Water Return Temperature
AHU Water Loop	Water Pressure
	Water Flow Rate
	Bypass Valve Opening
	Chilled Water Supply Temperature
Air System	Chilled Water Return Temperature
	Temperature
	Relative Humidity
	CO ₂ Concentration
	Air Flow Rate
	Air Damper Opening
OAE	Static Air Pressure
	Temperature
	Relative Humidity
	Air Flow Rate
	Static Air Pressure

In particular, we're interested in developing a digital twin of a HVAC system of the physical testbed. The structure of the air system in the testbed is shown in Figure 4. First the outdoor air is mixed with the return air from the room. Then the mixed air is passed through the Air Handling Unit (AHU) which consists of a cooling coil and fan. Next, the air will go through a heater and Variable Air Volume (VAV) to accurately control the air temperature and flow rate of the room. It may seem that the structure of the testbed is energy inefficient as the air goes through a cooling coil and a heater afterwards, but this is due to the testbed nature ensuring accurate and refine control of the room environment. Although real buildings do not have as refined control and sensory deployment as the testbed has, the testbed acts as a basis for creating DTs of the indoor built environment.

3.2 Digital Twin Model

We modeled the air system in Figure 4 with the hybrid approach. The Air Handling Unit (AHU), which contains a cooling coil and a fan, is modeled as a white box model. The electric heater and Variable Air Volume (VAV) are modeled together as a blackbox model. The properties of the room is modeled as a grey box model. The Proportional-Integral-Derivative (PID) feedback controller controlled the chilled water supply flow rate through the cooling coil to indirectly maintain the room set point temperature. The controller

¹<https://www.osisoft.com/pi-system>

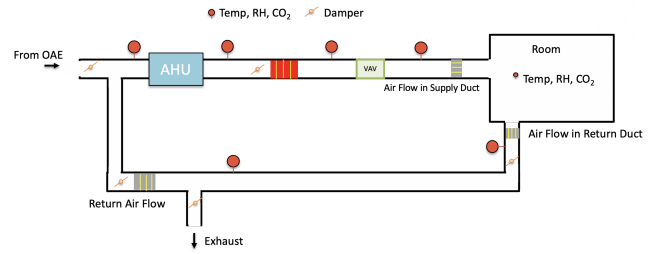


Figure 4: Air system of the testbed. The sensors placement is shown as a red pin point where temperature, relative humidity and CO₂ concentration are measured. The red box represents an electric heater. The Variable Air Volume (VAV) controls the air flow rate to the room.

is modeled as a white box model. The design choices are made by information available from the physical testbed and to reduce the number of uncertain parameters.

Figure 5 shows a block diagram of the connections and set-up of the components. The detailed equations used for each component are described in the following section.

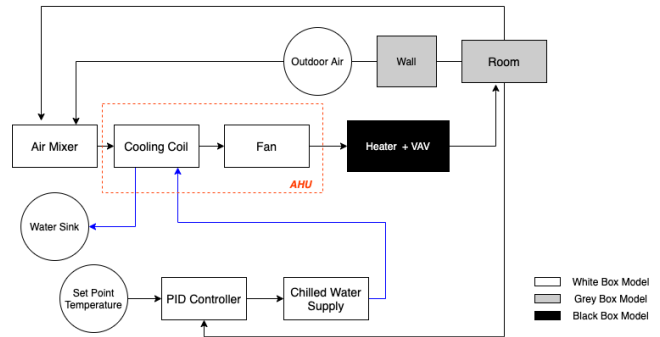


Figure 5: A block diagram that shows the white, grey, and black box model of each component in the air system of the room.

3.2.1 AHU. The AHU consists of a cooling coil and a fan. The cooling coil is modeled by applying convective heat transfer as shown in equation (1).

$$q = h_c A (T_s - T_a) \quad (1)$$

where q is the heat transfer (W), h_c is the convective heat transfer coefficient (W/m^2K), A is the heat transfer area of the surface (m^2), T_s is the temperature of the surface ($^{\circ}C$), and T_a is the temperature of the air ($^{\circ}C$). The mass flow rate is calculated by:

$$\dot{m} = \text{sign}(\Delta p) k \sqrt{|\Delta p|}, \quad (2)$$

where \dot{m} is the mass flow rate (kg/s) of a medium, k is the flow coefficient, and Δp is the pressure drop. The use of mass flow rate is to prevent the influence of density and temperature in the calculation.

3.2.2 Electric Heater and VAV. The electric heater and VAV are together modeled as a black box model. It is created through a two-hidden-layer neural network [18] that captures the relationship between the inputs and outputs of the components. ReLu function, which is defined as $relu(x) = \max(x, 0)$, was used to introduce non-linearity in the model. Given the input x , the output $f(x)$ is given by

$$f(x) = relu(w_1x + b_1) \cdot w_2 + b_2 \quad (3)$$

Parameters w_1, b_1, w_2, b_2 are learned during gradient descent. The ML model is trained with a month worth of past data and results in a mean-square error of 0.097. Figure 6 shows the difference between the prediction and the actual values in the test dataset.

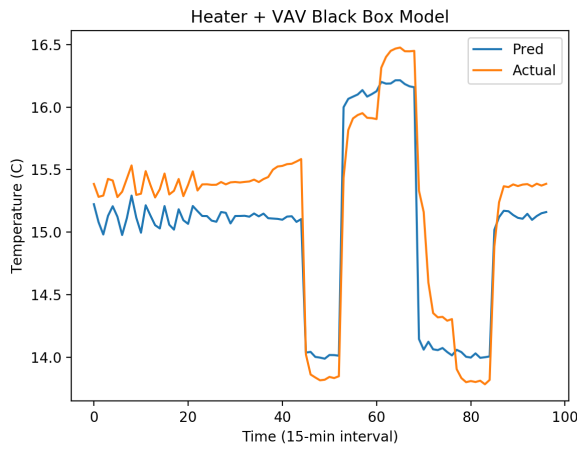


Figure 6: ML black box modeling the relationship between the input to the electric heater and output of the VAV.

3.2.3 Room. The room of the testbed is modeled as a volume of air and the wall connecting the testbed to the outside environment is modeled as an enveloped conductance value (grey box model). The conductance is estimated through a thermal circuit equation as shown in equation 4:

$$T - T_{out} = (h + TL + PL) \times R \quad (4)$$

where T is the room temperature ($^{\circ}C$), T_{out} is the outdoor temperature, h is the heat removed of the room (W), TL is the thermal load placed in the room, PL is the parasitic load (W), and R is the envelope wall resistance ($^{\circ}C/W$). The h is further calculated by applying heat balance equation:

$$h = c_p \rho q \Delta t \quad (5)$$

where c_p is the specific heat of the air, ρ is the air density, q is the air flow rate, and Δt is the difference between the return air temperature and the supply air temperature.

3.2.4 PID controller. The controller controls the chilled water flow rate that is supplied to the cooling coil. It continuously calculates the error values between the set point temperature and measured room temperature and applies corrections.

3.2.5 Summarizing. Table 3 shows the main parameters that are defined in the Modelica model. The list is not an exhaustive list. Other parameters such as cooling coil conductance and PID values of the controllers are not listed. The table only shows the major parameters that we are most concerned about.

Table 3: List of model parameters and their range.

Model Parameter	Symbol	min	max
Envelope Wall Conductance (W/K)	θ_1	100	300
Nominal Supply Air Temperature ($^{\circ}C$)	θ_2	13	16
Room Set Point Temperature ($^{\circ}C$)	θ_3	22	26
Nominal Outlet Air Temperature ($^{\circ}C$)	θ_4	25	30
Internal Heat Gain (W)	θ_5	100	1500
Nominal Cooling Load (W)	θ_6	1000	3000
Nominal Air Flow Rate (kg/s)	θ_7	0.1	0.3
Water Supply Temperature ($^{\circ}C$)	θ_8	6	8
Water Return Temperature ($^{\circ}C$)	θ_9	10	12
Nominal Water Mass Flow Rate (kg/s)	θ_{10}	2	4
Outdoor Temperature ($^{\circ}C$)	θ_{11}	22	26
Change in Water Pressure (Pa)	θ_{12}	30	100
Change in Air Pressure (Pa)	θ_{13}	500	1000

3.3 Experimental Design

We design a factorial experiment on three parameters: set point temperature, air supply temperature and thermal load for initial calibration. Factorial experiments are the most run efficient way to collect data and examine the relationship between different parameters. The heat load is set by using heaters. Heat load is used for simulating the occupancy in the room. 500 Watts to 1000 Watts of heat load corresponds to 5-10 people in the room, which is reasonable given the size of the testbed. Table 4 shows the parameters settings for the factorial experiment where A represents thermal load, B represents set point temperature, and C represents air supply temperature. There are 12 different combinations of settings and each is carried out twice with 2 hour duration. The duration is chosen to guarantee that the testbed will reach steady state. Therefore, there are a total of 24 experiments.

Table 4: Factorial Experiment Design with three independent variables: Thermal Load (A), Set Point Temperature (B), and Supply Air Temperature (C).

		$C_1(13^{\circ}C)$	$C_2(15^{\circ}C)$
$A_1(500W)$	$B_1(24^{\circ}C)$	Group 1	Group 2
	$B_2(26^{\circ}C)$	Group 3	Group 4
$A_2(750W)$	$B_1(24^{\circ}C)$	Group 5	Group 6
	$B_2(26^{\circ}C)$	Group 7	Group 8
$A_3(1000W)$	$B_1(24^{\circ}C)$	Group 9	Group 10
	$B_2(26^{\circ}C)$	Group 11	Group 12

The range of parameters are chosen based on typical temperature for the room. Each experiment was run for two hours, six

experiments were run in a day, and all the experiments were done in a period of 4 days in August 2020. The timeline is shown in Figure 7.



Figure 7: Timeline for a day of experiments.

3.4 Analysis

For simplifications, we analyze only the steady state of the experiments. Out of the 24 experiments we conducted, there are some experiments that do not converge to the corresponding set point temperature within two hours. The reason may be that the cooling coil exceeds its cooling capacity for high thermal load experiment. However, the purpose of the experiment is to perform initial calibration, so the 13 experiments that converge to set point temperature are enough for analysis. In equation 4, all parameters but the PL and R are known. We estimate the parasitic load and wall thermal resistance by minimizing the square error for the thermal circuit model. The objective is shown in equation 6, where $n = 13$.

$$\min \sum_{i=1}^n \left(\frac{T_i - T_{out,i}}{R} - h_i - TL_i - PL \right)^2 \quad (6)$$

Sensitivity analysis was also performed. The parameters in table 3 are used for analysis. The range of the parameters is chosen by using range of typical values that are used in the HVAC system in a hot climate area. The Morris method was applied with 13 parameters (θ_1 to θ_{13}), 80 trajectories and 12 levels.

The results of Morris method are evaluated by comparing the mean (μ) and standard deviation (σ) of each input parameters. Figure 8 shows a plot of μ^* against σ for each parameter. A higher μ^* value implies that the parameter is more sensitive to the output of the model, whereas a higher σ value implies that there are possible interactions with other parameters. Parameters that are not labeled on the plot are the ones that's closed to the bottom left of the graph (corresponds to nearly zero mean and zero standard deviation values). They are insignificant variables. The plot shows that nominal supply air temperature (θ_2), room set point temperature (θ_3), internal heat gain (θ_5), nominal cooling load of the coil (θ_6), water supply temperature (θ_8), water return temperature (θ_9), nominal water mass flow rate (θ_{10}) are the most influential parameters to the model. We can obtain parameters $\theta_2, \theta_3, \theta_8, \theta_9$ from the testbed. Therefore, we focus on tuning the rest of the parameters $\theta_5, \theta_6, \theta_{10}$.

Then we compare the cooling energy of the physical testbed and our virtual model. The result is shown in Figure 9. The plot shows that the cooling energy of the virtual system matches fairly well with the physical system.

Lastly, the communication between the sensor data and the model was done by utilizing the Functional Mockup Interface (FMI)², a standard co-simulation platform. Modelica model is converted to Functional Mockup Unit (FMU) and the continuous data stream is then fed into the FMU with a Python script.

²<https://fmi-standard.org/>

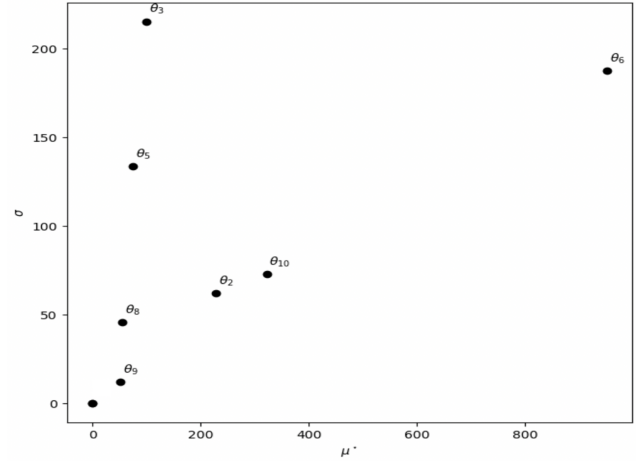


Figure 8: Graphical plot for sensitive measure μ^* and σ . The closer the parameters to the upper right, the more sensitive the parameter.

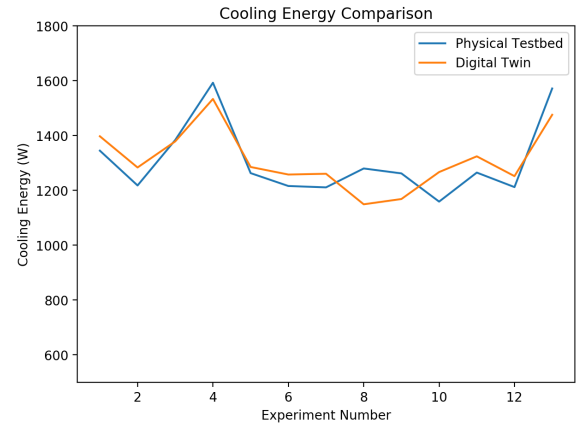


Figure 9: Comparison of cooling energy between the physical testbed and the digital twin model.

4 DISCUSSION

The limitations of the experiment are addressed as follows. Conducting the experiment under testbed conditions is increasingly facile compared to real life office spaces due to the substantial concentration of sensors in the testbed, relative to the average embedded office building. However, in real life, a more generalizable model must be realized in order to properly model a given physical entity with a much smaller sensor density. We use a testbed as a starting point to create a digital twin because of the large amount of sensors that are already deployed in the testbed. This aids in simplifying the experiment but can be extended to creating a digital twin in a real office building. For example, essential parameters can be identified beforehand and we limit the number of sensors that is needed to be deployed in a real office building to reduce costs and human effort.

In addition, we want to identify Modelica as a potent stepping stone usable for creating an actual digital twin.

Whether Modelica model can be easily scale or generalize is still ongoing research. Building a hybrid model with Modelica may be time consuming and also computationally expensive. However, it has a potential to be easily scale and generalize into multi-zone environments. The black box model was implemented through a simple 2-layer neural network structure. There may exist other architecture that can reach a higher prediction accuracy. However, the purpose for this research is to show the possibilities in applying ML in combination with physics-based model in Modelica. Our experiment focus on analyzing the steady state of the experiment, there may exist more deviations with transient state. Furthermore, just comparing the cooling energy between a physical entity and the DT model does not imperatively equate to the models accuracy or precision. Rather, the cooling energy comparison shown is indicative of Modelica-based hybrid approach as a supporting feature in digital twin development. Other metrics need to be developed to ensure that they are indistinguishable from one another.

5 CONCLUSION AND FUTURE WORK

In conclusion, Modelica shows potentials to act as a based in creating a hybrid DT model. It can be freely adjusted to match the level of abstraction that the physical entity can provide. The experiment shows the comparison of the cooling energy from the physical space and the virtual space matches fairly closely. However, there are still several challenges ahead. More experiments and testing need to be done to tune and validate the model. In order to realize the digital twin in a built environment, we need to develop a systematic metric to compare the digital model and physical entity. On top of that, the model needs to be generalized so that it is applicable to all the buildings.

The current model would be a launch point from which a fully generalized model will be realized. In future work, we plan to gradually scale the model recursively until the sensor comparisons encapsulate the entirety of the building architecture. In addition, we will dive deeper into physics-based machine learning model where we tried to embed physics equations into neural network architecture and compare it with the pure physics-based model and the hybrid model. A digital twin of a building helps engineers analyze and control the constructed environment more efficiently. It reduces significant amount of cost in energy usage and maintains occupants comfort. Moreover, it opens up new potential research avenues on the built environment.

ACKNOWLEDGMENTS

This research was funded by the Republic of Singapore's National Research Foundation through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (Sin-BerBEST) Program. BEARS has been established by the University of California, Berkeley as a center for intellectual excellence in research and education in Singapore. This research is also supported by CITRIS Seed Funding Program.

REFERENCES

- [1] P. Aivaliotis, K. Georgoulas, Z. Arkouli, and S. Makris. 2019. Methodology for enabling Digital Twin using advanced physics-based modelling in predictive maintenance. *Procedia CIRP* 81 (2019), 417–422. <https://doi.org/10.1016/j.procir.2019.03.072> 52nd CIRP Conference on Manufacturing Systems (CMS), Ljubljana, Slovenia, June 12–14, 2019.
- [2] A. Allouhi, Y. El Fouih, T. Kousksou, A. Jamil, Y. Zeraoui, and Y. Mourad. 2015. Energy consumption and efficiency in buildings: current status and future trends. *Journal of Cleaner Production* 109 (2015), 118 – 130. <https://doi.org/10.1016/j.jclepro.2015.05.139> Special Issue: Toward a Regenerative Sustainability Paradigm for the Built Environment: from vision to reality.
- [3] Yi Cai, Binil Starly, Paul Cohen, and Yuan-Shin Lee. 2017. Sensor Data and Information Fusion to Construct Digital-twins Virtual Machine Tools for Cyber-physical Manufacturing. *Procedia Manufacturing* 10 (2017), 1031–1042. <https://doi.org/10.1016/j.promfg.2017.07.094> 45th SME North American Manufacturing Research Conference, NAMRC 45, LA, USA.
- [4] California Public Utilities Commission. [n.d.]. Zero Net Energy. <https://www.cpuc.ca.gov/ZNE/>
- [5] Johnson Controls. 2019. *Applying Digital Twins to the Built Environment*. Technical Report. Johnson Controls. https://www.johnsoncontrols.com/-/media/jci/insights/2019/bts/jci-661_dv_digital_twin_white_paper_020819_6p_f3.pdf
- [6] Drury B. Crawley, Curtis O. Pedersen, Linda K. Lawrie, and Frederick C. Winkelmann. 2000. EnergyPlus: Energy Simulation Program. *ASHRAE Journal* 42 (2000), 49–56.
- [7] Abigail Francisco, Neda Mohammadi, and John E. Taylor. 2020. Smart City Digital Twin–Enabled Energy Management: Toward Real-Time Urban Building Energy Benchmarking. *Journal of Management in Engineering* 36, 2 (2020), 04019045. [https://doi.org/10.1061/\(ASCE\)ME.1943-5479.0000741](https://doi.org/10.1061/(ASCE)ME.1943-5479.0000741)
- [8] Peter Fritzson and Vadim Engelson. 1998. Modelica — A unified object-oriented language for system modeling and simulation. In *ECOP'98 — Object-Oriented Programming*, Eric Jul (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 67–90.
- [9] Michael Grieves and John Vickers. 2017. *Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems*. Springer International Publishing, Cham, 85–113. https://doi.org/10.1007/978-3-319-38756-7_4
- [10] Jon Herman and Will Usher. 2017. SALib: An open-source Python library for Sensitivity Analysis. *The Journal of Open Source Software* 2, 9 (jan 2017). <https://doi.org/10.21105/joss.00097>
- [11] Sakdirat Kaewunruen, Panrawee Rungskunroch, and Joshua Welsh. 2019. A Digital-Twin Evaluation of Net Zero Energy Building for Existing Buildings. *Sustainability* 11, 1 (2019). <https://doi.org/10.3390/su11010159>
- [12] Ioannis C. Konstantakopoulos, Andrew R. Barkan, Shiyong He, Tanya Veeravalli, Huihan Liu, and Costas Spanos. 2019. A deep learning and gamification approach to improving human-building interaction and energy efficiency in smart infrastructure. *Applied Energy* 237 (2019), 810 – 821. <https://doi.org/10.1016/j.apenergy.2018.12.065>
- [13] Max D. Morris. 1991. Factorial Sampling Plans for Preliminary Computational Experiments. *Technometrics* 33, 2 (1991), 161–174. <http://www.jstor.org/stable/1269043>
- [14] The European parliament and the council of the European Union. 2018. Directive (EU) 2018/2002 of the European Parliament and of the Council of 11 December 2018 amending Directive 2012/27/EU on energy efficiency (Text with EEA relevance.). *Official Journal of the European Union* (2018).
- [15] Colin Parris. 2017. A Twin-building Army: GE previews first-ever Digital Twin Analytics Workbench. <https://www.linkedin.com/pulse/twin-building-army-ge-previews-first-ever-digital-twin-colin-parris/>. Accessed: 2020-02-03.
- [16] J. Poon, I. C. Konstantakopoulos, R. Arghandeh, P. Jain, J. F. Fisac, S. S. Sastry, S. K. Panda, C. Spanos, and S. R. Sanders. 2015. FailSafe: A generalized methodology for converter fault detection, identification, and remediation in nanogrids. In *2015 IEEE International Conference on Building Efficiency and Sustainable Technologies*. 73–78.
- [17] Kevin Prawiranto, Jan Carmeliet, and Thijs Defraeye. 2021. Physics-Based Digital Twin Identifies Trade-Offs Between Drying Time, Fruit Quality, and Energy Use for Solar Drying. *Frontiers in Sustainable Food Systems* 4 (2021), 286. <https://doi.org/10.3389/fsufs.2020.606845>
- [18] Stuart Russell and Peter Norvig. 2009. *Artificial Intelligence: A Modern Approach* (3rd ed.). Prentice Hall Press, USA.
- [19] Benjamin Schleich, Nabil Anwer, Luc Mathieu, and Sandro Wartzack. 2017. Shaping the digital twin for design and production engineering. *CIRP Annals* 66, 1 (2017), 141 – 144. <https://doi.org/10.1016/j.cirp.2017.04.040>
- [20] Siemens. 2017. For a digital twin of the grid: Siemens solution enables a single digital grid model of the Finnish power system. <https://assets.new.siemens.com/siemens/assets/api/uuid:09c20834-4ed4-49d8-923d-ebcc541cab37/inno2017-digitaltwin-e.pdf>
- [21] Robert Stackowiak. 2019. *Azure IoT Solutions Overview*. Apress, Berkeley, CA, 29–54. https://doi.org/10.1007/978-1-4842-5470-7_2

- [22] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee. 2019. Digital Twin in Industry: State-of-the-Art. *IEEE Transactions on Industrial Informatics* 15, 4 (2019), 2405–2415. <https://doi.org/10.1109/TII.2018.2873186>
- [23] F. Tao and M. Zhang. 2017. Digital Twin Shop-Floor: A New Shop-Floor Paradigm Towards Smart Manufacturing. *IEEE Access* 5 (2017), 20418–20427. <https://doi.org/10.1109/ACCESS.2017.2756069>
- [24] Willow Team. 2019. Unlocking a new era of smart buildings through the digital twin. <https://www.willowinc.com/2019/10/18/unlocking-a-new-era-of-smart-buildings-through-the-digital-twin/>. Accessed: 2020-02-03.
- [25] Urvashi Verma. 2018. Enertiv's smart building digital twin saves building owners 20%. <https://inbuildingtech.com/smart-buildings/digital-twin-smart-building/>. Accessed: 2020-02-03.
- [26] Michael Wetter, Wangda Zuo, Thierry S. Nouidui, and Xiufeng Pang. 2014. Modelica Buildings library. *Journal of Building Performance Simulation* 7, 4 (2014), 253–270. <https://doi.org/10.1080/19401493.2013.765506> arXiv:<https://doi.org/10.1080/19401493.2013.765506>
- [27] Jared Willard, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. 2020. Integrating Physics-Based Modeling with Machine Learning: A Survey. arXiv:2003.04919 [physics.comp-ph]
- [28] Yu Yang, Guoqiang Hu, and Costas J. Spanos. 2019. HVAC Energy Cost Optimization for a Multi-zone Building via a Decentralized Approach. arXiv:arXiv:1905.10934