MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

# A physics-constrained deep learning framework for dynamic modeling of vapor compression systems

Ma, JiaCheng; Dong, Yiyun; Qiao, Hongtao; Laughman, Christopher R.

TR2025-137 October 01, 2025

**Abstract**

Data-driven dynamic models typically offer faster execution than their physicsbased counterparts described by large systems of nonlinear differential-algebraic equations (DAEs) with quantitatively reasonable accuracy. Therefore, development of such models can be extremely useful for design optimization, controls, fault detection and diagnostics of vapor compression based building Heating, ventilation and air conditioning (HVAC) systems. As the complexity and scale of vapor compression systems (VCS) increase rapidly across the industry, a modular approach of generating and interconnecting data-driven component models enables model reuse and efficient adaption to arbitrary system layouts. Despite the flexibility, the modular integration for system model generation can suffer from nonphysical behaviors of violating conservation laws due to inevitable prediction errors associated with each component. This paper presents a data-driven modeling framework that exploits state-of-the-art deep learning techniques for constructing component models, while enforcing physical conservation for system simulations. A general-purpose system solver is developed to handle arbitrary configurations by automatically integrating data-driven or interchangeable physics-based component models into a system model. Results of an air-source heat pump system reveal a significant speedup with good agreement, compared to high-fidelity first-principles models.

*Applied Energy 2025*



Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

# A physics-constrained deep learning framework for dynamic modeling of vapor compression systems

Jiacheng Ma[a], Yiyun Dong[b], Hongtao Qiao[c,*], Christopher R. Laughman[c]

[a]*Ray W. Herrick Laboratories, School of Mechanical Engineering, Purdue University, West Lafayette, IN, USA*
[b]*Department of Physics, University of Washington, Seattle, WA, USA*
[c]*Mitsubishi Electric Research Laboratories, Cambridge, MA, USA*

**Abstract**

Data-driven dynamic models typically offer faster execution than their physics-based counterparts described by large systems of nonlinear differential-algebraic equations (DAEs) with quantitatively reasonable accuracy. Therefore, development of such models can be extremely useful for design optimization, controls, fault detection and diagnostics of vapor compression based building Heating, ventilation and air conditioning (HVAC) systems. As the complexity and scale of vapor compression systems (VCS) increase rapidly across the industry, a modular approach of generating and interconnecting data-driven component models enables model reuse and efficient adaption to arbitrary system layouts. Despite the flexibility, the modular integration for system model generation can suffer from nonphysical behaviors of violating conservation laws due to inevitable prediction errors associated with each component. This paper presents a data-driven modeling framework that exploits state-of-the-art deep learning techniques for constructing component models, while enforcing physical conservation for system simulations. A general-purpose system solver is developed to handle arbitrary configurations by automatically integrating data-driven or interchangeable physics-based component models into a system model. Results of an air-source heat pump system reveal a significant speedup with good agreement, compared to high-fidelity first-principles models.

*Keywords:* Transient simulation, Deep learning, Gated recurrent unit,

*Corresponding author.
*Email address:* qiao@merl.com (Hongtao Qiao)

Vapor compression system, Component-based modeling, Physical conservation

## Nomenclature

### Symbols

| | |
|---|---|
| $\dot{m}$ | Mass flow rate [kg/s] |
| $\dot{Q}$ | Heat transfer rate [J/kg] |
| $A$ | Area [$m^2$] |
| $E$ | Internal energy [J] |
| $f$ | Factor [−] |
| $h$ | Specific enthalpy [J/kg] |
| $M$ | Mass [kg] |
| $P$ | Power [W] |
| $p$ | Pressure [Pa] |
| $T$ | Temperature [K] |

### Greek letters

| | |
|---|---|
| $\eta$ | Efficiency [−] |
| $\omega$ | Compressor speed [Hz] |
| $\phi$ | Valve opening [−] |
| $\rho$ | Density [$kg/m^3$] |

### Subscript

| | |
|---|---|
| $a$ | Air |
| $amb$ | Ambient |
| $cond$ | Condenser |

| | |
|---|---|
| $dis$ | Discharge |
| $e$ | Refrigerant state close to exit |
| $evap$ | Evaporator |
| $i$ | Refrigerant state close to inlet |
| $in$ | Inlet |
| $is$ | Isentropic |
| $lat$ | Latent |
| $loss$ | Heat loss |
| $out$ | Outlet |
| $r$ | Refrigerant |
| $suc$ | Suction |
| $sys$ | System |
| $tot$ | Total |
| $v$ | Volumetric |

## 1. Introduction

A staggering 40% of building energy consumption can be attributed to heating, ventilation and air conditioning (HVAC) systems (Energy, 2010) where vapor compression systems (VCS) serve as key components for space conditioning. With the emerging demand for adopting energy-efficient VCS equipment for HVAC systems in residential and commercial building sectors, design optimization and control for performance improvement of VCS are vital in reducing energy utilization and achieving a sustainable future. Development of accurate and computationally affordable dynamic models of such systems serves an important role in this process by facilitating system performance analysis, design and evaluation of control, fault detection and diagnostics (FDD) algorithms, etc.

Dynamic modeling of VCS has been an active research area over past decades (Rasmussen, 2012; Li et al., 2014). The modeling paradigms appearing in the literature can be roughly categorized by the level of physical information embraced to develop a model as white-box, grey-box, and black-box. While white-box models employ physical conservation laws and black-box models, contrarily, are identified in a data-driven fashion, the most commonly seen models fall under the category of gray-box. This type of models is largely built upon physical laws, but incorporates empirical correlations to simplify certain descriptions. In the following context, these models are referred to as physics-based models, in order to draw a distinction from black-box or data-driven models.

To capture the complex thermo-fluid behaviors of VCS that dominantly reside in two-phase heat exchangers (HX), physics-based models are described by a set of governing conservation equations, i.e., mass, energy, and momentum. A solution scheme for these models generally involves spatial discretization and nonlinear algebraic coupling, which results in systems of nonlinear differential-algebraic equations (DAEs) (Laughman and Qiao, 2018; Chakrabarty et al., 2021). The two dominant discretization schemes, namely finite volume (FV) and moving boundary (MB), have been extensively studied and found appropriate for different applications. The FV approach is able to capture exhaustive flow behaviors spatially within HX at a cost of simulation speed due to high dimensionality, while the MB method features a low-order formulation but suffers from a lack of robustness as a result of inherent discontinuities (Bendapudi et al., 2008; Qiao et al., 2016; Kim et al., 2021; Ma et al., 2021). Despite successes in predicting transient behaviors

of various VCS configurations using physics-based models through experimental validations (Qiao et al., 2015b; Ma, 2024), the applicability of these physics-based dynamic models for control and FDD purposes continues to encounter remarkable challenges regarding computational speed, robustness and initialization as the complexity and scale of system configurations tend to increase rapidly in the near future (Zhang et al., 2019).

In contrast, data-driven models constitute a solution path regarding challenges for a variety of real-time applications, as they omit vastly the underlying physics of the phenomena within components. Constructing data-driven surrogate models for high-fidelity physics-based models in model-based design applications has gained significant interest recently (Loka et al., 2023). With advances in scientific machine learning capability and enrichment of large-scale experimental data sets, most models have been investigated for steady-state performance prediction of VCS and their components, e.g. Shao et al. (2012); Yousaf et al. (2023); Wang et al. (2024), with a very limited number of studies focusing on characterizing the nonlinear dynamic behaviors. Among them, Habtom (1999) pioneered employing feedforward neural networks fed with delayed output signals for the identification of a refrigeration system using samples collected on a laboratory setup to predict temperature and humidity inside the system. Following a similar approach, Yoon and Lee (2010) developed neural network models trained by experimental data to capture the dynamics of an air-to-water heat pump system where past control inputs to the system augmented with model outputs were fed into the model for predictions. Chen and Fu (2020) proposed a Long Short-Term Memory (LSTM) model, to predict the performance of an electrical vehicle air conditioner. Chen et al. (2022) subsequently explored an implementation of integrating recurrent neural network (RNN) based heat exchanger models and a multilayer perceptions (MLPs) compressor model into an air conditioner system model. However, their data-driven component models were implicitly coupled at the training stage through shared input variables (e.g., compressor speed was used as an input to both compressor and heat exchanger model development) which hindered their reusability for different system configurations.

It appears to be a common practice in existing studies to develop a single data-driven model for capturing the dynamics of an entire system neglecting interactive behaviors of components, or component models within a system that are confined to a specific system layout. As a consequence, a model generated in this manner is solely valid for a specific system of interest under

certain operating conditions, without any ease in modifying system configurations and reusing it for broader purposes. Furthermore, training of a system-level model can be prohibitively expensive for large-scale system architectures such as variable refrigerant flow (VRF) systems that consist of multiple indoor units under a wide range of operating conditions, even though a number of identical components are often employed within the system and can be efficiently represented by the same component model. A modular approach is therefore attractive for general-purpose model development. Creating and integrating data-driven component models or a combination of data-driven and physics-based models in a hybrid fashion offers the flexibility to reuse readily trained models for arbitrary cycle configurations. Moreover, the importance of enforcing physical conservation laws arises concerning the integration of data-driven component models for a system model in presence of inevitable prediction errors associated with each component model Hansen et al. (2023). Without a care of physical conservation constraints, data-driven models could carry out unrealistic predictions and suffer from numerical failures.

Despite a great amount of research efforts in dynamic VCS modeling, there exists room for further advancements in modeling approaches concerning simulation accuracy, efficiency and robustness. Clearly a literature screening indicates an absence of a general modeling framework for VCS that can incorporate physics-constrained data-driven modeling techniques for constructing component models and consequently system models for arbitrary configurations. This paper aims to complete the research gap of a physics-constrained data-driven modeling framework and broaden research interests regarding high performance computational tools for dynamic VCS modeling. The state-of-the-art deep learning methods were exploited to construct component models with generalized interfaces for physical conservation. After that, a system solver was employed to complete a system model that can integrate data-driven and physics-based component models interchangeably in a hybrid fashion. To the authors' knowledge, this is the first demonstration of a novel physics-constrained hybrid modeling framework within the field. In summary, the proposed dynamic modeling framework offers appealing features including

- Physical conservation enforcement: The design of fast, accurate and robust data-driven component models involves superimposing constraints of physical conservation laws that lead to physically conserving system

model aggregation.

- Flexible model adaption: Generalized interfaces are adopted for deep learning component models under categories of heat exchanger and mass-flow device, which allows interchanges between data-driven and physics-based model deployment as needed.

- Modular model integration: A general-purpose system solver is proposed to seamlessly integrate component models for arbitrary cycle configurations and enable model reuses.

The remainder of this paper is structured as follows. Section 2 describes methodologies of component modeling and system model integration. Section 3 presents a case study where data-driven models are generated for a dual compressor air-source heat pump (ASHP) system and compared with physics-based Modelica models to demonstrate the efficacy of the proposed approach. Section 4 summarizes conclusions of the present paper, followed by discussions in Section 5.

## 2. Methodologies

### *2.1. Data-driven dynamic modeling overview*

The modularity of deep learning dynamic vapor compression system (VCS) modeling can be realized by constructing models that capture the behavior of each individual component and then integrating them following the physical conservation laws. Since the dominant dynamics reside in two-phase heat exchangers (HX), autoregressive time-series prediction modeling approaches can be employed to capture the complex behavior, while mass-flow devices such as compressors and valves are effectively modeled using feedforward formulations with nonlinear function mappings under quasi-steady-state assumptions. Depending on the input/output arrangements for deriving component models, solution procedures of an integrated system model may involve numerical iterations to determine intermediate variables, at every time step when system dynamics are simulated to evolve on time.

Key elements of the proposed modeling framework are outlined as follows:

- Feedforward static models for mass-flow devices using nonlinear performance mappings;

- Time-series prediction models for two-phase head exchangers using recurrent neural networks constrained by physical conservation laws;
- A system solver designed to handle arbitrary configurations of data-driven models, capable of formulating and solving residual equations to systematically determine intermediate variables and make predictions orderly.

### *2.2. Feedforward neural network static component models*

#### *2.2.1. Compressor*

Development of quasi-steady-state compressor models is carried out as a multi-input-multi-output (MIMO) mapping task. In general, variable-speed compressor models take inputs of the refrigerant suction state, discharge pressure and actuation signals, while predicting mass flow rate, power and discharge temperature or enthalpy as outputs. Black-box models (e.g., 10-coefficient polynomials) rely on a large set of data for training, and usually suffer from non-physical behaviors and poor extrapolation performance outside the operating envelope that is specified to generate the data. Therefore, instead of direct input-output mapping, physical knowledge can be incorporated by mapping from inputs to efficiencies that describe compression processes including volumetric and isentropic efficiencies (Yang et al., 2009). To complete a compressor model, an energy balance can be formed across the compressor, which is characterized by a heat loss factor to account for heat losses to the ambient. Consequently, feedforward neural networks are exploited to define a mapping $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$ with

$$\boldsymbol{y} = \begin{bmatrix} \eta_v & \eta_{is} & f_{loss} \end{bmatrix}^T \in \mathbb{R}^3, \tag{1}$$

$$\boldsymbol{x} = \begin{bmatrix} \omega & p_{suc} & h_{suc} & p_{dis} \end{bmatrix}^T \in \mathbb{R}^4 \tag{2}$$

where outputs consist of volumetric efficiency, isentropic efficiency and heat loss factor, model inputs are compressor speed, suction pressure, suction enthalpy and discharge pressure. After that, compressor performance can be predicted utilizing those neural network model outputs and readily available inputs. The refrigerant mass flow rate is calculated by

$$\dot{m} = \omega \eta_v \rho_{suc} V_s \tag{3}$$

where $V_s$ is the displacement of compressor volume and $\rho_{suc}$ is the suction density. The power consumption is determined by

$$P = \frac{\dot{m}(h_{dis,is} - h_{suc})}{\eta_{is}} \tag{4}$$

where $h_{dis,is}$ is the discharge enthalpy assuming an isentropic compression process. Afterwards, the discharge enthalpy can be obtained by

$$h_{dis} = h_{suc} + \frac{P - \dot{Q}_{loss}}{\dot{m}} \tag{5}$$

where the heat loss is obtained as a ratio to the power input using the predicted heat loss factor $\dot{Q}_{loss} = f_{loss}P$. It is important to note that by predicting these dimensionless efficiencies using neural network models, the overall compressor model is applicable to other refrigerants since thermodynamic property evaluations are absent during training given a dataset of inputs and outputs specified in Eq. (1) and (2). In addition, compressor model outputs can be bounded to physically feasible values by limiting predicted efficiencies to valid ranges, which is essential to robust system simulations when coupled to other component models.

*2.2.2. Expansion valve*

Generating expansion valve models follows a similar approach as compressors. A feedforward neural network is trained to predict a combined term of discharge coefficient $C_d$ and valve opening area $A$ that is often unavailable to model development with model inputs of inlet pressure $p_{in}$, inlet enthalpy $h_{in}$, outlet pressure $p_{out}$ as well as a normalized valve opening signal $\phi$ which determines the valve opening area by $A = A_{min} + \phi(A_{max} - A_{min})$. The refrigerant mass flow rate through the valve can subsequently be predicted by adopting a typical throttling process description

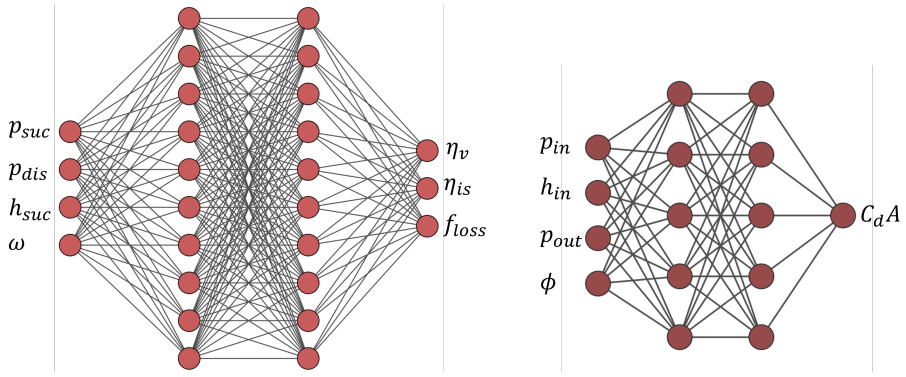
$$\dot{m} = C_d A \sqrt{2\rho_{in}(p_{in} - p_{out})}. \tag{6}$$

Furthermore, the expansion process is assumed to be enthalpic, leading to $h_{out} = h_{in}$. To summarize, a mapping for expansion valve models is specified as

$$\boldsymbol{y} = \begin{bmatrix} C_d A & h_{out} \end{bmatrix}^T \in \mathbb{R}^2 \tag{7}$$

$$\boldsymbol{x} = \begin{bmatrix} p_{in} & h_{in} & p_{out} & \phi \end{bmatrix}^T \in \mathbb{R}^4. \tag{8}$$

Fig. (1) outlines feedforward neural networks employed in the present work for compressor and expansion valve modeling.




(a) (b)

Figure 1: Data-driven mass flow device models using feedforward neural networks: (a) compressor; (b) expansion valve.

### *2.3. Recurrent neural network heat exchanger modeling*

Considering a fixed-step time discretization, a general description of HX dynamics can be presented in a time-invariant form

$$\boldsymbol{y}_t = \Phi(\boldsymbol{u}_{0:t}, \boldsymbol{x}_0) \tag{9}$$

where $\boldsymbol{y}_t$ denotes observable outputs at time instance $t$, $\boldsymbol{x}_0$ denotes the initial condition of internal states, $\boldsymbol{u}_{0:t}$ represents a sequence of inputs that manipulate those states and $\boldsymbol{\Phi}(\cdot)$ is an unknown nonlinear map governing system behaviors. Without access to the internal states, data-driven models seek a nonlinear approximator $\boldsymbol{f}(\cdot)$ that can predict outputs $\boldsymbol{y}_t$ depending on historical values of inputs and outputs to certain orders

$$\boldsymbol{y}_t = \boldsymbol{f}(\boldsymbol{y}_{t-n_y:t-1}, \boldsymbol{u}_{t-n_u:t}) \tag{10}$$

where a time interval of 1 second is utilized here and in the following context for illustration.

Apparently, the selection of $n_y$ and $n_u$ is of critical importance to the performance of resulting models, as a trade-off between prediction accuracy and model complexities. Apart from that, choices of input and output variables play a vital role in model prediction capabilities. Regarding heat exchanger as a generic control volume, input-output responses are typically captured by the time evolution of the refrigerant and the secondary fluid states at the inlet and outlet of flow paths. Fig. 2 depicts those variables for a fin-and-tube

HX commonly employed in an ASHP system, where inlet mass flow rate and enthalpy, back pressure, and upstream enthalpy at the outlet in a reverse-flow scenario are featured as inputs for the refrigerant flow, while inlet conditions, flow rate and ambient pressure are considered for the air side. Correspondingly, outputs of a HX model include the refrigerant states at infinitesimal points on edges of the HX inlet and exit as well as exit temperature and heat capacities of the air flow.

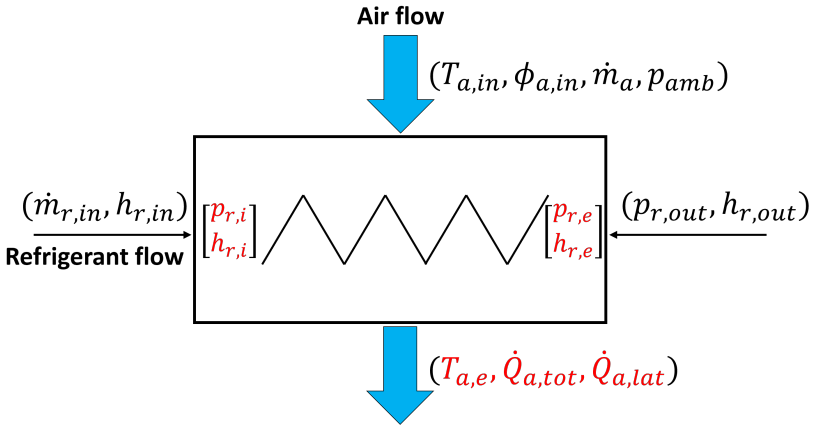



Figure 2: Overview of input and output variables commonly adopted for data-driven HX modeling that may violate conservation laws (output variables are marked in red).

This setup may initially look adequate concerning the predictive performance of a heat exchanger component model, however, issues of violating conservation laws arise when such models are integrated into a system. Consider an ASHP system comprised of $n$ heat exchangers and $m$ mass-flow devices. An energy balance can be formed with regard to the entire system as

$$\frac{dE_{sys}}{dt} = \sum_{k=1}^{n} \dot{Q}_k + \sum_{k=1}^{m} (P_k - \dot{Q}_{k,loss}) \tag{11}$$

where $E_{sys}$ represents the internal energy of the entire system, $\dot{Q}_k$ represents the air-side capacity of each heat exchanger following the sign convention that inflow energy (e.g., for evaporator) is positive and outflow is negative, $P_k$ and $\dot{Q}_{k,loss}$ are power input and heat loss of each mass-flow device, respectively. Since capacities, power and heat loss are predicted separately by

individual component models, their balance can not be enforced at steady-state conditions due to inevitable prediction errors, consequently violating the energy conservation stated in Eq. (11). To tackle this issue, the internal energy of a heat exchanger is incorporated as a model input and updated subject to a forward difference scheme with a time step $\Delta t$

$$E_{k,t+1} = E_{k,t} + (\dot{m}_{k,in} h_{k,in} - \dot{m}_{k,out} h_{k,out} + \dot{Q}_k)\Delta t \quad k = 1, 2, \ldots, n \tag{12}$$

which describes the heat exchanger energy balance in a discrete-time form. Usually air-side mass and energy storage are negligible for cross-flow heat exchangers, which leads to a quasi-steady-state model description. As a result, heat exchanger internal energy is composed of energy storage in the refrigerant and metal walls. Following this scheme, the total energy source for a system at every time step can be obtained by summing up those of heat exchangers as shown in Eq. (12), since mass flow devices are modeled as static components without energy storage

$$\Delta E_{tot} = \sum_{k=1}^{n} (\dot{m}_{k,in} h_{k,in} - \dot{m}_{k,out} h_{k,out} + \dot{Q}_k). \tag{13}$$

Along the refrigerant loop, enthalpy flows across mass flow devices can be represented equivalently using power and heat loss of those components according to the energy balance as revealed in Eq. (5), while intermediate quantities in between heat exchangers are canceled out. As a consequence, the total energy source obtained from heat exchanger models corresponds to that for the entire system appearing in Eq. (11), which indicates that under the current setup system energy conservation is respected at steady-state conditions as heat exchanger models evolve to $E_{k,t+1} = E_{k,t}$.

A similar argument can be made for system mass conservation. Of equal importance to realize system energy balance is to ensure a consistent refrigerant charge prediction for the overall system. It can be easily understood that predicting the refrigerant charge residing in a heat exchanger as a model output leads to an inconsistent total charge prediction over time due to inevitable prediction errors associated with each heat exchanger model. Therefore, the refrigerant charge is considered as an input to a heat exchanger model and is updated according to a discrete-time mass balance (Dong et al., 2024)

$$M_{k,t+1} = M_{k,t} + (\dot{m}_{k,in} - \dot{m}_{k,out})\Delta t \tag{14}$$

which reveals that the total refrigerant charge across the system remains constant $\sum_{k=1}^{n} M_{k,t+1} = \sum_{k=1}^{n} M_{k,t}$ once models are initialized, since interface mass flow rates are canceled out at every time step.

As per the above discussion, the ultimate sets of input and output variables selected for heat exchanger model development in Eq. (10) are outlined

$$\boldsymbol{u} = \begin{bmatrix} T_{a,in} & \phi_{a,in} & \dot{m}_a & p_{amb} & \dot{m}_{r,in} & h_{r,in} & h_{r,out} & p_{r,out} & M_r & E_r \end{bmatrix}^T \in \mathbb{R}^{10}, \tag{15}$$

$$\boldsymbol{y} = \begin{bmatrix} p_{r,i} & h_{r,i} & p_{r,e} & h_{r,e} & T_{a,e} & \dot{Q}_{a,tot} & \dot{Q}_{a,lat} \end{bmatrix}^T \in \mathbb{R}^{7}. \tag{16}$$

It is essential to note that this setup is generally applicable to the development of mass and energy conserving data-driven heat exchanger models, invariant to time-series forecasting modeling techniques. This work exploits capability and efficiency of recurrent neural networks (RNN) in identifying the complicated thermo-fluid dynamics. Specifically, gated recurrent unit (GRU) as a particular RNN architecture invented to mitigate the vanishing gradient problem is considered (Rehmer and Kroll, 2019). GRU is specialized for learning long-term dependencies within sequential data among deep learning models, and has gained growing interest in the identification of nonlinear dynamical systems (Bhattacharya et al., 2022; Jordan et al., 2021). An encoder-decoder sequence-to-sequence GRU architecture is adopted in the present work. As shown in Fig. 3, sequences of input features and outputs tracked with a look-back window of context length $N$ are augmented as an input context sequence $\boldsymbol{x}_{T-N:T-1}$ to the multi-layer GRU network. An encoder GRU block processes the input sequence and then a decoder GRU block generates the predicted output sequence subject to a given length. In terms of a HX model, the output sequence length is set to one since further predictions can only be made until the cycle model is solved at the current time step. Each of the encoder and decoder blocks consist of 2 GRU layers in this work. For each layer, the GRU computes the hidden state $\boldsymbol{h}_t$ given a vector element $\boldsymbol{x}_t$ of an input sequence by

$$\boldsymbol{r}_t = \sigma(\boldsymbol{W}_{ir}\boldsymbol{x}_t + \boldsymbol{b}_{ir} + \boldsymbol{W}_{h}\boldsymbol{h}_{t-1} + \boldsymbol{b}_{hr}) \tag{17}$$

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}_{iz}\boldsymbol{x}_t + \boldsymbol{b}_{iz} + \boldsymbol{W}_{hz}\boldsymbol{h}_{t-1} + \boldsymbol{b}_{hz}) \tag{18}$$

$$\boldsymbol{n}_t = \tanh\left(\boldsymbol{W}_{in}\boldsymbol{x}_t + \boldsymbol{b}_{in} + \boldsymbol{r}_t \odot (\boldsymbol{W}_{hn}\boldsymbol{h}_{t-1} + \boldsymbol{b}_{hn})\right) \tag{19}$$

$$\boldsymbol{h}_t = (1 - \boldsymbol{z}_t) \odot \boldsymbol{n}_t + \boldsymbol{z}_t \odot \boldsymbol{h}_{t-1} \tag{20}$$

where $\boldsymbol{r}_t$, $\boldsymbol{z}_t$, $\boldsymbol{n}_t$ are the reset, update and new gates, respectively. $\boldsymbol{W}$ denotes weight matrices and $\boldsymbol{b}$ denotes bias vectors. $\sigma(\cdot)$ is the element-wise sigmoid

function and $\odot$ is the Hadamard product. $\boldsymbol{h}_{t-1}$ is the hidden state at time $t-1$ or zero for the initial hidden state. The input element at a time $\boldsymbol{x}_t$ differs depending on the GRU layer and block. For instance, $\boldsymbol{x}_t$ represents each element of the input sequence $\boldsymbol{x}_{T-N:T-1}$ for the first layer of the encoder block, or the hidden state $\boldsymbol{h}_t$ of the first layer for the second layer. The computation proceeds to the prediction phase once the hidden state $\boldsymbol{h}_{T-1}$ of the last encoder layer is obtained, which is fed to the decoder as the initial hidden state. Since the decoder GRU block is set to generate predictions for a single time step $T$ as mentioned above, the one-step input sequence $\boldsymbol{x}_T = \begin{bmatrix} \boldsymbol{u}_T^T & \boldsymbol{y}_{T-1}^T \end{bmatrix}^T$ of the context sequence is fed as inputs to the first layer of the decoder, where the final hidden state $\boldsymbol{h}_T$ is subsequently passed through a set of fully connected layers consisting of a linear layer, a rectified linear unit (ReLU) layer followed by another linear layer that map the hidden state to outputs $\boldsymbol{y}_T$, which concludes predictions at the current time step. This procedure is then repeated by updating the context sequence with input features for the next time step $\boldsymbol{u}_{T+1}$ and the predicted outputs $\boldsymbol{y}_T$.

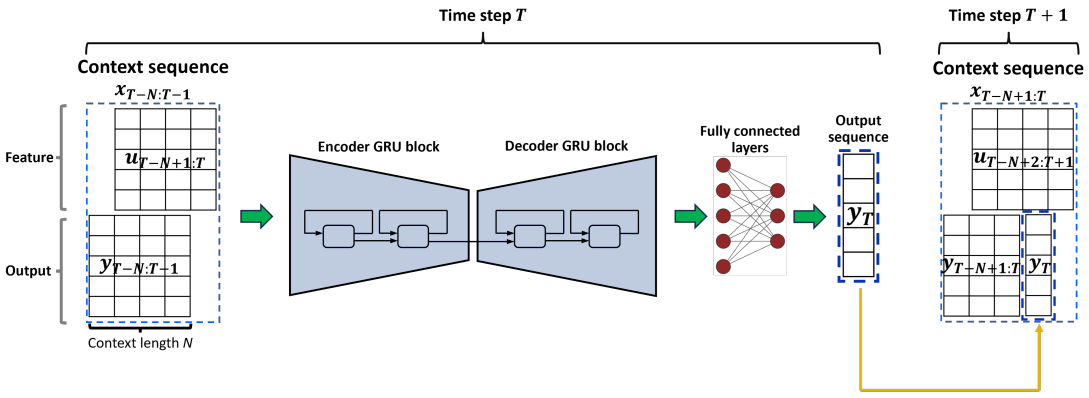



Figure 3: An illustration of the GRU architecture for HX model with one-step prediction.

Training the aforementioned model involves first processing time-series data of feature and output trajectories as context and output sequences. A sliding window of length $N+1$ is applied to the raw transient data to truncate it into shortened pieces spanning $[T-N, T]$ where $\boldsymbol{u}_{T-N+1:T}$ and $\boldsymbol{y}_{T-N:T-1}$ are augmented to form a context sequence and the training target $\boldsymbol{y}_T$ is used to compute the losses with model predictions. The entire dataset of transient pieces is then split into training, validation and testing sets, respectively. The mean squared error (MSE) is adopted to measure the training loss of each

truncated piece, which then adds up to the total loss. The learning rate is adjusted by a scheduler during training. After each training epoch, the loss over the validation set is evaluated to adapt the learning rate by a reduction factor when it stops decreasing for a certain number of consecutive epochs.

### *2.4. System solver*

While each component model is characterized by a uniquely selected set of input/output variables to ensure individual model performance, a system solver is necessary for modular-based dynamic modeling to orderly integrate component models and progress model predictions over time. As continuity equations are formed when interconnecting component models, the refrigerant states at junctions remain unknown at every time step and involve numerical iterations for solutions when connected components share the same boundary conditions. Specifically, the refrigerant pressure is considered a boundary condition/input at ports of mass-flow devices and the nominal exit port of heat exchangers for data-driven models derived before, which leads to a pressure iteration variable when connecting a heat exchanger exit port to a mass-flow device. In general, a system solution scheme proceeds as follows:

1. Identify iteration variables at component connections, i.e. intermediate pressures, and form the minimum number of residual equations for arbitrary system configurations before online simulations;
2. Solve residual equations at every time step during simulations to enforce continuity;
3. Execute all component models and progress computations for a prediction interval.

It can be easily seen that residual equations should be formed when a heat exchanger outlet is connected to a mass-flow device inlet or when two mass-flow devices are connected. Concerning multi-branch fluid loops where some components serve as splitters or mergers, four configurations can be recognized with a need to solve for unknown pressures due to continuity constraints, as listed in Fig. 4.

When a system model is composed, the system layout and component connections can be encoded as a directed graph and represented by a matrix for further analysis where information about connections between a row component and a column component can be extracted from the corresponding entry. Although the coding rule is ambiguous, it is required to allow acausal

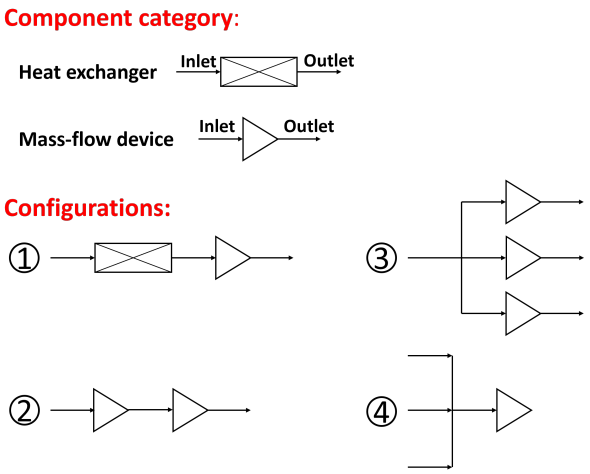



Figure 4: Four configurations to form residual equations for solving intermediate pressures: 1. a heat exchanger connected to a mass-flow device; 2. a mass-flow device connected to a mass-flow device; 3. a splitter component connected to branches of all mass-flow devices; 4. components connected to a mass-flow device merger.

connections and distinguish between connection patterns including outlet-inlet, inlet-outlet, inlet-inlet, and outlet-outlet. For instance, the present work employs a rule that an outlet-inlet connection is coded 1, an outlet-outlet connection is coded 2, and an inlet-inlet connection is coded 3. Take the system shown in Fig. 5 as an example for illustration, where component models are loaded and then connected in an acausal way such that not all connections are explicitly carried out as outlet-inlet. Following the coding rule, a matrix is generated to reveal connection patterns

$$\begin{bmatrix} 0 & 3 & 0 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{21}$$

Recall that the ultimate goal is to automatically form residual equations to solve iteration variables within the system. To realize that it is beneficial to convert all component connections into outlet-inlet formats from the information encoded in the above matrix. Given entries equal to 2 or 3 in the matrix, a breadth-first search (BFS) is performed to locate the merger

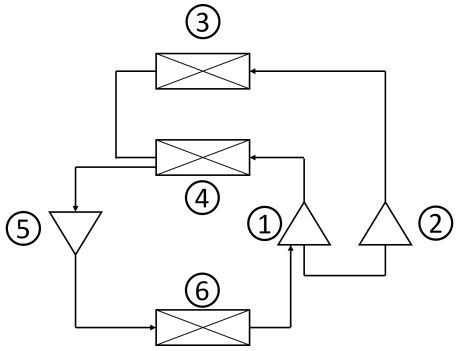



Figure 5: Component models connected in an acausal way to form a system.

or splitter that those connected components share. Then the entries are converted to 1 if they follow an outlet-inlet pattern and 0 otherwise. As a consequence, an adjacency matrix can be formed where entries of 1 indicate that the outlet of the row component is connected to the inlet of the column component

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{22}$$

The adjacency matrix reveals abundant information on system layout and component connections. Essentially, mergers and splitters can be identified by

$$\text{component k is a merger:} \quad \sum_{j=1}^{N} a_{j,k} > 1 \,\&\, \sum_{j=1}^{N} a_{k,j} = 1, \tag{23}$$

$$\text{component k is a splitter:} \quad \sum_{j=1}^{N} a_{k,j} > 1 \,\&\, \sum_{j=1}^{N} a_{j,k} = 1 \tag{24}$$

assuming that a component cannot serve as a merger and splitter at the same time. The task of generating residual equations is then equivalent to comparing the connection pattern of each component to the four configurations shown in Fig. 4 to seek matches. In terms of the illustrative case, a

system of two residual equations is formed as depicted in Fig. 6 to solve for two junction pressures and enforce mass balances, which are decoupled in this specific case but may not be in general. During online simulations, the residual equations system

$$\boldsymbol{F}\left(\begin{bmatrix} p_a \\ p_b \end{bmatrix}\right) = \begin{bmatrix} \dot{m}_6(p_a) - \dot{m}_1(p_a) - \dot{m}_2(p_a) \\ \dot{m}_5(p_b) - \dot{m}_3(p_b) - \dot{m}_4(p_b) \end{bmatrix} = \mathbf{0} \tag{25}$$

can be solved by multivariate root-finding algorithms such as Powell's method (Kochenderfer and Wheeler, 2019).

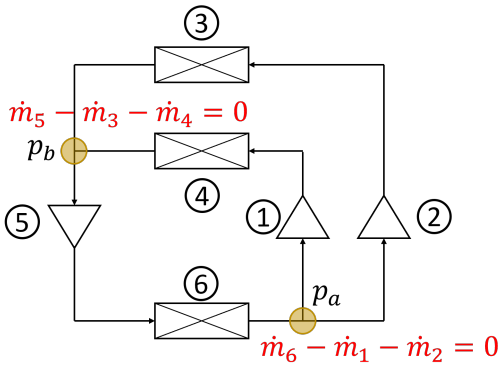



Figure 6: System layout with connections converted to outlet-inlet format and residual equations generated to solve for intermediate pressures.

Once the system solver completes the configuration analysis and proceeds to the prediction stage, the GRU heat exchanger models are executed to obtain outputs at the time step based upon given inputs and past information. Afterwards, the readily formulated residual equations are solved using predicted refrigerant states. Intermediate pressures solved by iterations are used to evaluate mass-flow device models and update input variables for heat exchangers to make predictions for the next time step. Note that among those inputs listed in Eq. (15), the refrigerant charge and heat exchanger internal energy are updated following the forward difference scheme stated in Eq. (12) and (14). Consequently, the system model remains in the form of a discrete-time dynamical system. Fig. (7) summarizes workflows of the system solver.

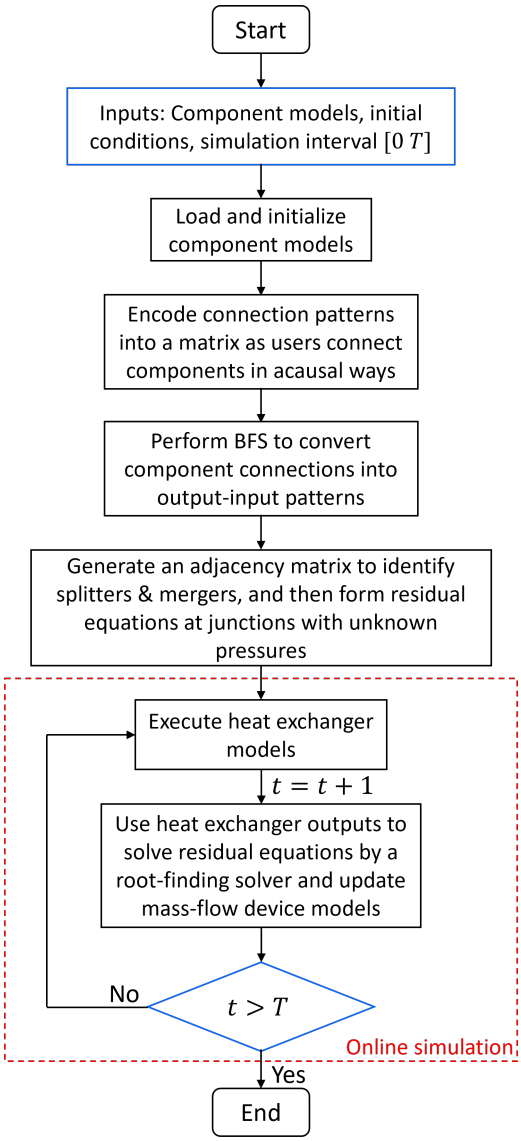



Figure 7: Workflows of the system solver.

## 3. Case study: an air-source heat pump

This section presents a case study of the proposed data-driven dynamic modeling framework for an air-source heat pump system. High-fidelity physics-based component models are constructed and implemented using the Modelica language and the Dymola environment. Each component model is simulated using randomly generated input profiles to capture model behaviors corresponding to feasible input space. The simulated data is then utilized for training data-driven models according to methodologies described in Section 2. Performance of the integrated data-driven system model is evaluated with regard to simulation speed and prediction accuracy.

### *3.1. System description*

A dual-compressor air-source heat pump system comprising two parallel compressors, two parallel condensers, an electronic expansion valve (EXV) and an evaporator is considered for demonstration. Physics-based component models are built upon conservation laws incorporated with empirical correlations and performance maps. The compressor and EXV models are similar to those described in Section 2 except for the fact that efficiency maps are fitted with high-order polynomials. Dynamic heat exchanger modeling is carried out using a finite volume method where a staggered grid is exploited to discretize governing equations of mass, momentum and energy for the refrigerant flow while the refrigerant pressure and enthalpy are selected as state variables. Coil metal structures including tubes and fins are modeled as lumped capacitances with one-dimensional heat conduction, while the air side employs quasi-steady-state analysis assuming uniform surface temperatures. Refer to Qiao et al. (2015a) for a comprehensive description of the model development. It is important to note that static momentum balances which relate pressure differences to frictional pressure drops by algebraic equations are adopted. The frictional pressure drop is correlated using a power law, which leads to the static momentum balance

$$\Delta p = K \Delta p_0 \big(\frac{\dot{m}}{\dot{m}_0}\big)^{\alpha} \tag{26}$$

for each control volume where $\Delta p_0$ denotes the pressure drop at a nominal condition corresponding to a mass flow rate $\dot{m}_0$, $K$ and $\alpha$ are fitted coefficients to account for various operating conditions. When a heat exchanger model is coupled to mass-flow device models, the exit mass flow rate is calculated from the pressure difference given a heat exchanger back pressure

according to Eq. (26). Therefore, the back pressure should be iterated to satisfy the continuity constraint as illustrated in Eq. (25) at each time step. Moreover, the heat exchanger model is formulated to capture spatial variations along coil circuitry by dividing each tube into several control volumes and connecting them based on the actual multi-row coil circuitry configurations. As a result, a high-dimensional differential algebraic equation system is formed for a heat exchanger model. Fig. 8 reveals the Modelica model that describes the ASHP system of interest. In the present study, compressor speeds and EXV opening are actuated to drive system transients in heating operations, while fan speeds and ambient conditions including inlet air temperature and humidity for each heat exchanger are fixed. The condenser inlet air temperature, relative humidity, and volume flow rate are 20°C, 60%, and $0.4\,\mathrm{m^3\,s^{-1}}$, respectively, while the evaporator air-side inputs are 7°C, 87%, and $0.72\,\mathrm{m^3\,s^{-1}}$.

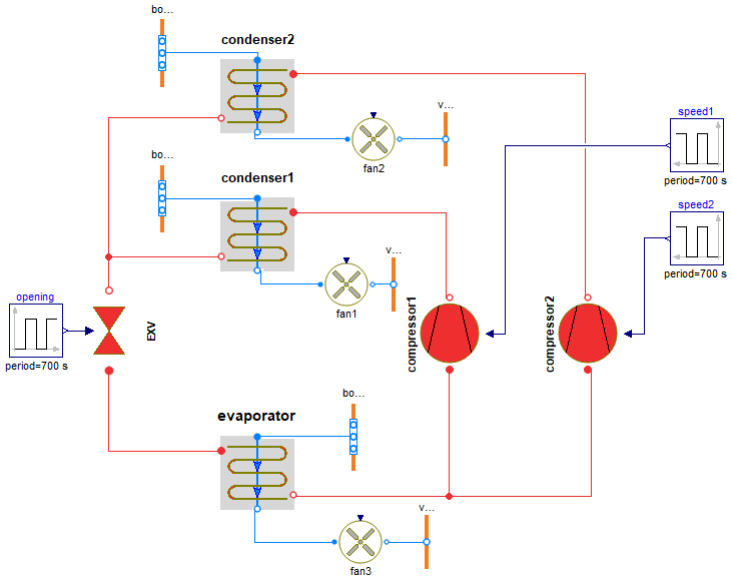



Figure 8: Modelica model of the dual-compressor ASHP system.

### *3.2. Generating data-driven models*

The primary goal of this study is to generate fast and robust surrogate models to the high-fidelity physics-based models for VCS that can expedite design and control development. Since the data-driven model setup involves

physical quantities that are readily available from physics-based model predictions but may not be easily accessible through measurements, the discussion of model training is focused on utilizing simulation data instead of experimental data. Despite the fact that simulating the entire system can be prohibitively slow due to the high-dimensional differential equation system and closure relations of component coupling, executing a component model driven by boundary conditions for data generation is practical for training purposes.

Physics-based heat exchanger models are excited by boundary condition profiles to collect trajectories of model predictions. In order to capture model behaviors both under steady-state and transient conditions, an input profile is constructed with consecutive step perturbations where the step length is selected so that the model can reach steady-state before boundary conditions are perturbed again. Concerning those on the refrigerant side, profiles of the inlet mass flow rate and enthalpy, outlet enthalpy and pressure are generated by performing random walks to extensively explore the input space, aiming to improve the generalization capability of trained models. Based on the operating envelope of the system, upper and lower bounds are set for each input variable as $\boldsymbol{u}_{max} = \begin{bmatrix} \dot{m}_{in,max} & h_{in,max} & h_{out,max} & p_{out,max} \end{bmatrix}^T$ and $\boldsymbol{u}_{min} = \begin{bmatrix} \dot{m}_{in,min} & h_{in,min} & h_{out,min} & p_{out,min} \end{bmatrix}^T$. The magnitude of inputs at each step is determined by random walks along the input space

$$\boldsymbol{u}[k+1] = \boldsymbol{u}[k] + \boldsymbol{Z}\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_4) \tag{27}$$

where $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_4)$ is a vector of standard Gaussian random numbers, and the diagonal matrix $\boldsymbol{Z}$ with elements of the vector $(\boldsymbol{u}_{max} - \boldsymbol{u}_{min})/l$ on its diagonal controls magnitudes of each step change for input variables where the constant factor $l$ scales variable ranges. The generated path of each input is then converted to a profile of step changes and bounded to feasible values. 500 steps of random walk are carried out and input profiles with a step length of 30s are generated to simulate condenser and evaporator models. Fig. 9 illustrates the workflow of generating training data for heat exchanger models using input profiles. The time-series model predictions are obtained with a sampling interval of 1 second. Trajectories of the predicted refrigerant charge and coil internal energy are augmented with input profiles to obtain the training input data as specified in Eq. (15), along with predicted output variables specified in Eq. (16) to complete the training dataset.

When implementing the proposed GRU heat exchanger model, apparently

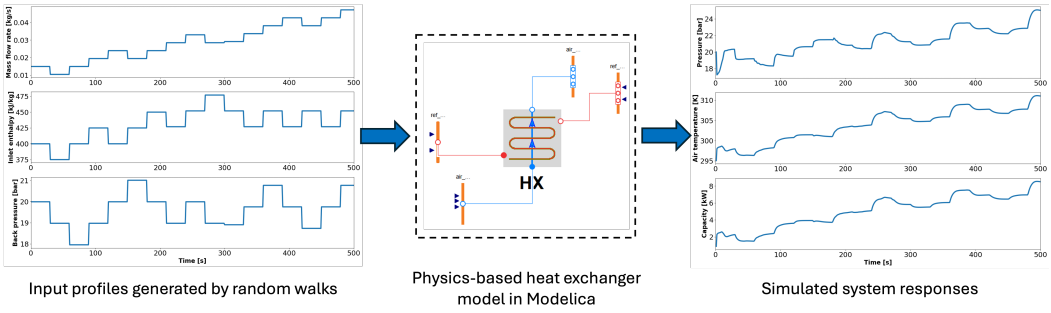



Figure 9: Workflow of generating training data for heat exchanger models.

the input sequence context length plays a crucial role in model performance as a trade-off between predictive capability and computational efficiency is involved, since longer time period dependencies residing in input sequences generally improve prediction accuracy, at a cost of computation time. Furthermore, a long context length increases complexities for model initialization. A context length of 2 is selected in the present study through a trial and error procedure. That means model outputs tracked back 2 seconds, features of the previous step along with those of the current step are fed into the GRU model as a context sequence to predict current output variables. The hidden size of each GRU layer is set to 256, and an initial learning rate of $2 \times 10^{-4}$ is applied. Each heat exchanger model is trained for 50 epochs with a batch size of 64.

Collecting training data for static models of mass-flow devices is approached in a straightforward way by randomly generating samples in the input space without a need for random walk, since outputs of these models do not embrace time dependency. A feedforward neural network of 2 hidden layers and 10 neurons per layer is adopted for the compressor model with the sigmoid activation function. The network is trained using $10^4$ samples for 1000 epochs with a batch size of 128. The performance of the trained model is evaluated by the root mean squared error (RMSE), which yield $7 \times 10^{-6}$ kg/s, 1.6 W, and 167 J/kg for outputs of mass flow rate, power, and discharge enthalpy, respectively. Training of the EXV model follows the same process where a network of 2 hidden layers and 5 nodes per layer is found appropriate for valve mass flow rate prediction. An RMSE of $3 \times 10^{-5}$ kg/s is revealed when testing the trained model.

All of the models are implemented using PyTorch with a CPU, and

trained using an Adam optimizer. Before training, each dataset is normalized by linearly scaling every feature to the range [0 1]. All computations were carried out on a typical desktop computer.

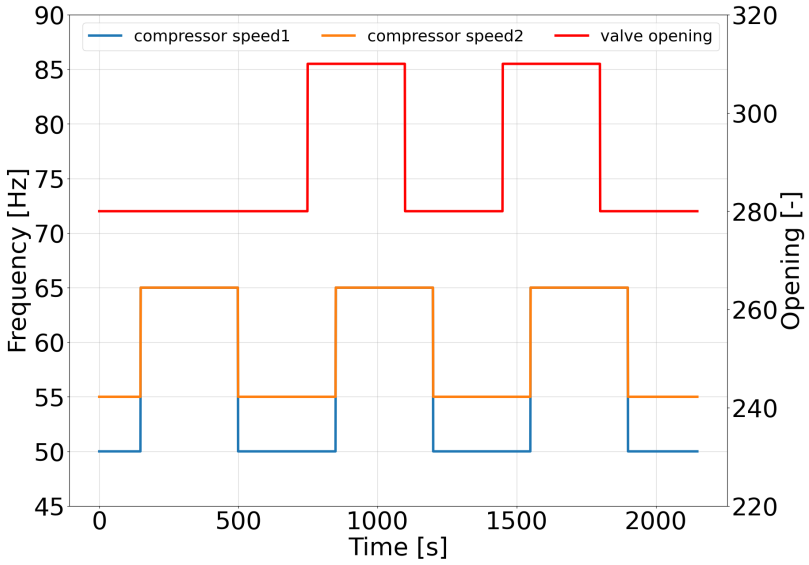



Figure 10: Compressor and valve actuation for system transients.

### *3.3. Simulation results*

Component models described above are integrated into a system model to capture load-change transients of the ASHP system. Since the system configuration can be represented as shown in Fig. 6, two residual equations are formed by the system solver to obtain junction pressures at a merger and a splitter with regard to the layout. Fig. 10 depicts the actuation of compressor speeds and EXV opening, where the valve opening is controlled by a step motor with a range of 500 steps. Two compressors are actuated with pulse signals of different magnitudes. A pulse signal staggered with those of compressors is applied to the valve opening, driving the system dynamics between different operating conditions.

To examine the performance of the proposed models, simulation results are compared with physics-based models in Dymola in terms of accuracy and simulation speed. Fig. 11 reveals predicted heat exchanger inlet pressures. The system is initialized at a steady state with low compressor speeds and a small valve opening. When both compressors are operated at high speeds,

the evaporating pressure becomes lower and condenser pressures slightly increase. As the EXV opens more when compressor speeds are at low values, the evaporating pressure rises which leads to increases in condenser pressures as well. Shortly after this valve actuation, compressor speeds return to a high level and drive the system to another operating condition. As a result, the pressure ratio increases. When the valve opening is switched to the initial position, the evaporating pressure and condenser pressures all decrease accordingly, after which the system behaviors are repeated as actuation signals go through the same combinations. It can be seen from the figure that the integrated system model based on neural network component models (denoted NN in the figure) reproduces system responses as the physics-based model with good agreement.

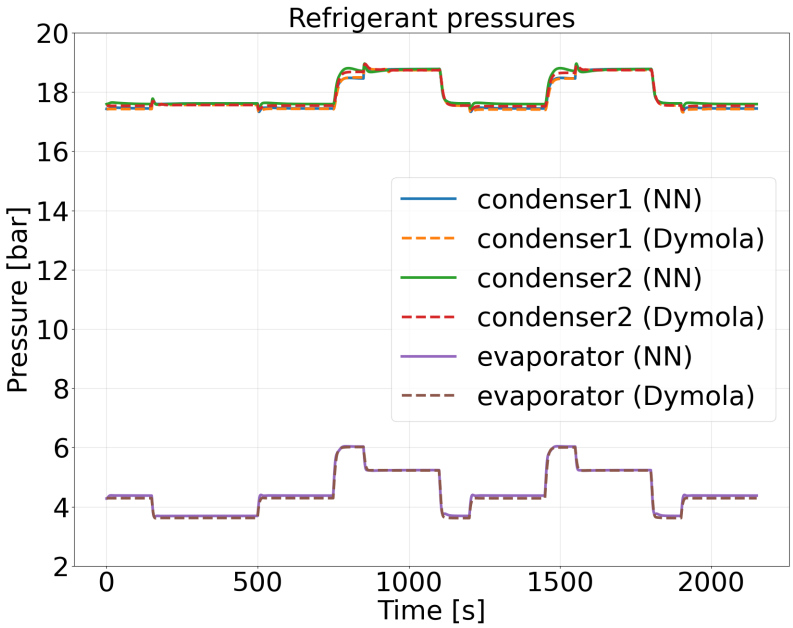



Figure 11: Comparisons of refrigerant pressures.

Fig. 12 reports the evolution of refrigerant mass flow rates. The liquid mass flow rate increases corresponding to increases of compressor speeds or the valve opening. However, the vapor mass flow rates on the compressor side exhibit different trends. Despite the increase of total vapor mass flow rate at high compressor speeds, the ratio of two flow paths changes when the system converges to steady-state conditions. As compressor speeds are equal,

the refrigerant flow redistribution from the evaporator results in a decrease in flow rate for compressor 2, while a significant increase for compressor 1. Similarly, when the total mass flow rate decreases, the compressor with a relatively higher speed yields a larger ratio and thus exhibits an increase in the flow rate. It is revealed that the data-driven model captures dominant dynamics and agrees well with the Modelica model, though steady-state error is observed, potentially due to prediction errors associated with mass-flow device models.

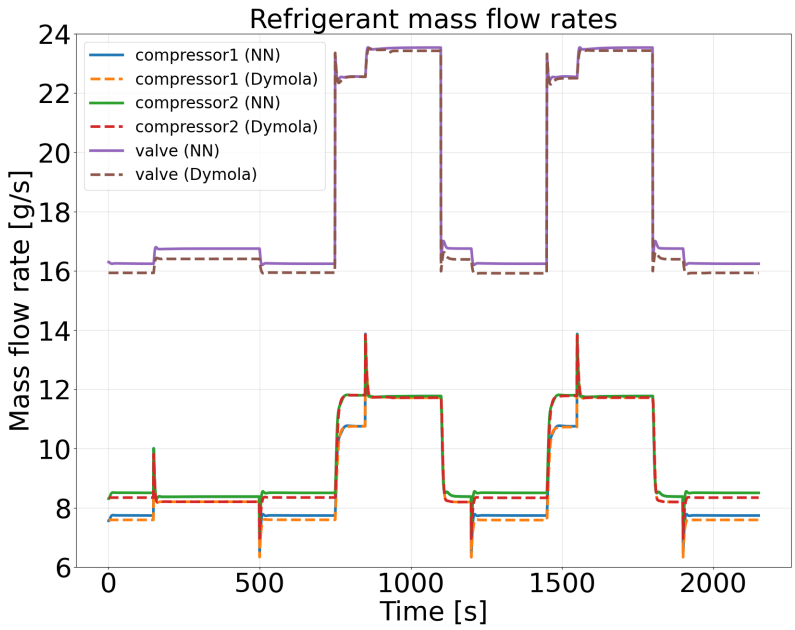



Figure 12: Comparisons of refrigerant mass flow rates.

Fig. 13 reveals the refrigerant charge migration of each heat exchanger. Generally, the refrigerant migrates to the evaporator when the EXV is widely opened, while more refrigerant resides in condensers at high compressor speeds. Similar to the results of mass flow rates, redistribution of the refrigerant charge among two condensers can be observed. When compressor speeds increase to the same level, though refrigerant migrates from the evaporator, the amount of refrigerant residing in condenser 1 becomes smaller as the refrigerant is equally distributed at the steady state when two condensers operate under identical conditions. The evaporator charge prediction responds well with that of the Modelica model, while discrepancies for con-

densers can be attributed to prediction errors in mass flow rates and the forward difference update scheme utilizing a fixed time step. However, of equal importance to the prediction accuracy is physical conservation properties of the system model. To verify mass conservation, the sum of refrigerant charge in three heat exchangers is shown in the figure as well. It reveals that the total refrigerant charge is completely determined at model initialization, and remains constant thereafter, aligning with the model setup for mass conservation. It should be noted that the data-driven model and the Modelica model reveal the same total charge prediction since identical initial conditions were applied in both simulations.

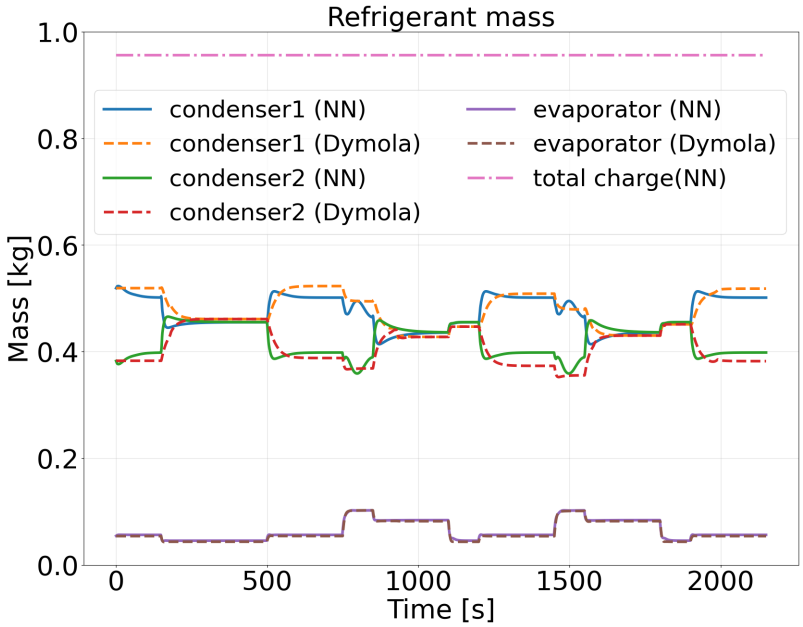



Figure 13: Comparisons of refrigerant charge.

Heat exchanger air-side capacity and the total compressor power are shown in Fig. 14. It can be seen that the compressor power generally follows the actuation of compressor speeds, and heat exchanger capacities embrace impacts of the refrigerant pressures and mass flow rates. The maximum heating capacity is obtained at a condition of high compressor speed and large valve opening that leads to a large mass flow rate. Simulation results indicate that behaviors of power consumption and capacities are well captured by the system model following an excellent agreement with the Modelica counter-

part. On top of that, energy flows in and out of the system are tracked and depicted in the figure to verify energy balances at steady-state conditions. The cycle inflow energy consists of compressor power consumptions as well as heat transfer from the ambient to the evaporator

$$E_{in} = P_{tot} + \dot{Q}_{a,evap} \tag{28}$$

where $P_{tot}$ denotes total power consumption of two compressors, and $\dot{Q}_{a,evap}$ denotes the evaporator air-side capacity. Likewise, the cycle outflow energy includes heat delivery by condensers and the compressor heat loss

$$E_{out} = \dot{Q}_{a,cond} + \dot{Q}_{loss}. \tag{29}$$

Note that the energy flow quantities are not equal during transients due to system energy storage. However, they must be balanced at a steady state with respect to the energy conservation law. As shown in the figure, the cycle inflow and outflow energy yield good agreement at all steady-state conditions, strongly supporting the energy conserving characteristic of the proposed modeling framework. In addition, results of heat exchanger air exit temperatures reveal similar trends as capacities, which are shown in Fig. 15.

The prediction accuracy of data-driven models is compared with physics-based models and quantified using the mean absolute percentage error (MAPE), which measures discrepancies between two time-series data sets as

$$\text{MAPE}(\boldsymbol{y}_{1:n}, \hat{\boldsymbol{y}}_{1:n}) = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{y_t - \hat{y}_t}{\hat{y}_t} \right| \times 100\% \tag{30}$$

where $\boldsymbol{y}_{1:n}$ represents data-driven model predictions and $\hat{\boldsymbol{y}}_{1:n}$ represents those of the Modelica model in this case. MAPE of the refrigerant pressures, air-side capacities, air exit temperatures, and total compressor power consumption are reported in Fig. 16. Relatively large values are observed in the evaporating pressure and coil capacities. However, results of all quantities lie below 2%, indicating that the data-driven models can achieve a similar level of accuracy as high-fidelity models.

To simulate the system load-change operation spanning 2150 seconds considered in this study, the data-driven system model completes in 40.4 seconds, while 146.8 seconds for the Modelica model, yielding a 3.6 times speedup by the developed model.

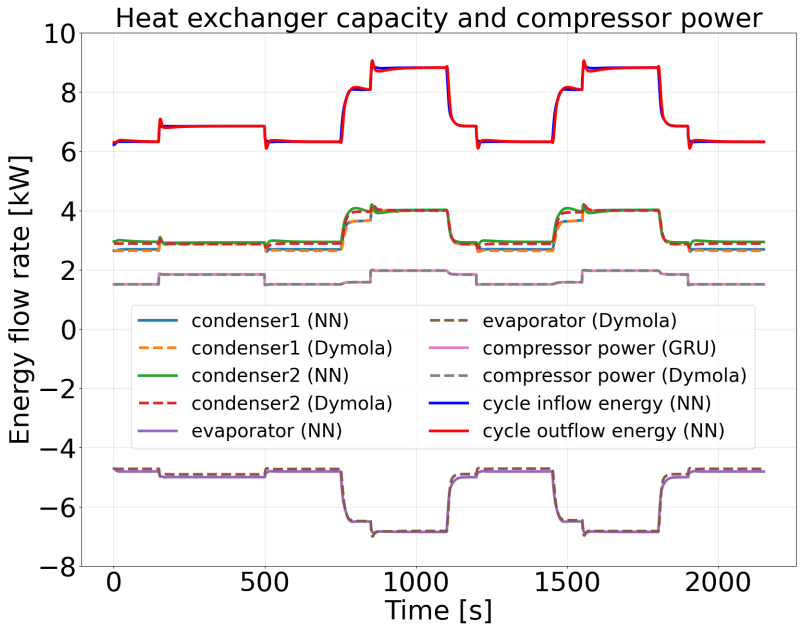



Figure 14: Comparisons of heat exchanger capacity and compressor power.

## 4. Conclusions

Despite a wealth of powerful data-driven modeling techniques, particularly in the rapidly evolving field of deep learning, there remains a noticeable absence of guidance for their application to vapor compression systems, or thermo-fluid systems generally, in an efficient and physically meaningful manner. This paper aims to bridge the gap by proposing a general-purpose deep learning framework for capturing the complicated two-phase flow dynamics of such systems. A modular implementation of component models is adopted to provide the flexibility of model reuses for accommodating arbitrary system configurations. Dynamic heat exchanger models based on the gated recurrent unit and feedforward neural network models for mass-flow devices are developed using simulation data of high-fidelity physics-based models and integrated into a system model. Then a system solver is implemented to analyze arbitrary layouts of component connections and progress transient simulations. A common challenge shared by data-driven models is the potential violation of physical conservation laws in predictions with a lack of physical constraints to enforce those principles. To tackle it, this work proposes mass and energy conserving heat exchanger models that en-

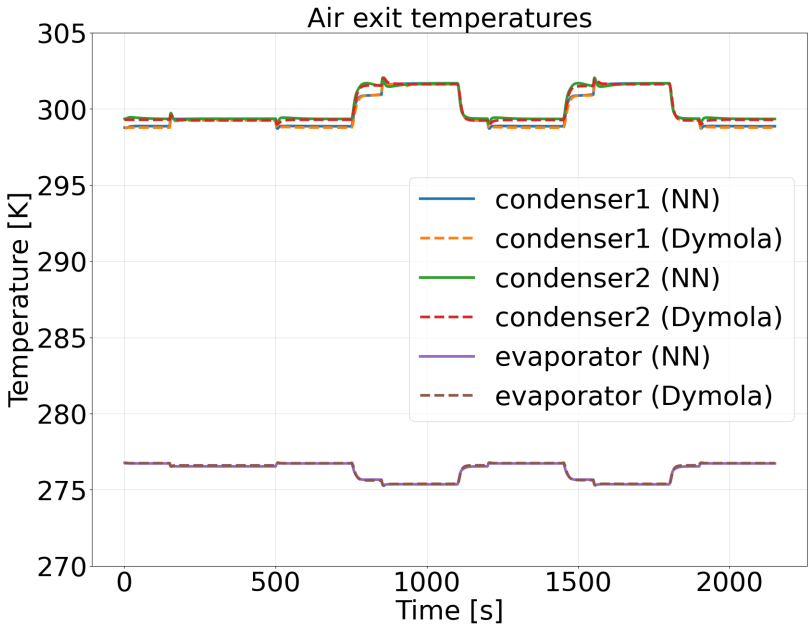



Figure 15: Comparisons of air exit temperatures.

sure a consistent system charge throughout transients and energy balances at steady-state conditions.

Data-driven component models are constructed for a dual-compressor heat pump system to capture system behaviors under load-change transients. Performance of the integrated system model is evaluated and compared with the physics-based counterpart in Modelica. Simulation results reveal good agreements between the two models and a 3.6 times speedup of the data-driven system model in terms of computational speed. The proposed approach can be attractive to the development and assessment of controls, fault detection and diagnostics for a variety of thermo-fluid systems.

## 5. Discussions

The overall performance of data-driven dynamic models meets the requirement of fast and accurate predictions in capturing dominant system behaviors. This section intends to spark discussions on further improvement of model development towards model-based control and FDD applications.

- Though this study employs recurrent neural networks for dynamic heat exchanger modeling, it is important to note that the model setup and

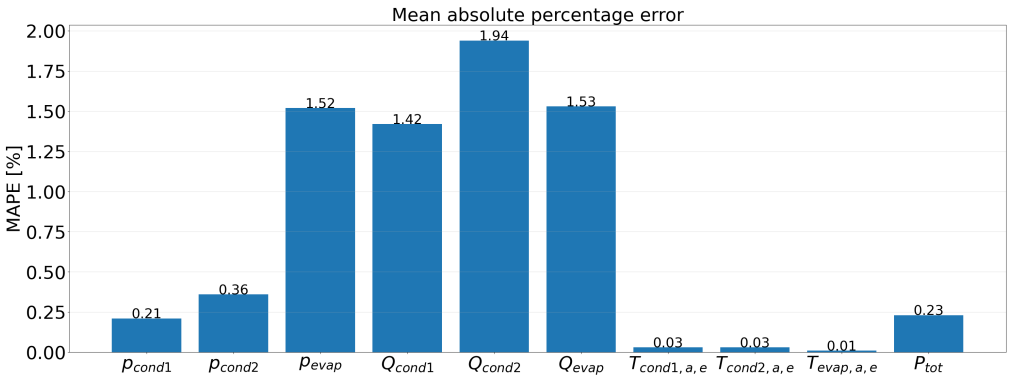



Figure 16: Mean absolute percentage error (MAPE): condenser1 inlet pressure; condenser2 inlet pressure; evaporator inlet pressure; condenser1 air-side capacity; condenser2 air-side capacity; evaporator air-side capacity; condenser1 air exit temperature; condenser2 air exit temperature; evaporator air exit temperature; total compressor power.

interfaces are generally applicable, invariant to a specific data-driven modeling technique and can be extended to model other dynamic thermal system components such as accumulators. With the selection of input/output variables and the update scheme for respecting conservation laws, other time-series foresting methods are worth investigating to capture heat exchanger dynamics. Similarly, nonlinear mapping approaches or different neural network architectures can be investigated to model mass-flow devices. Moreover, since the accuracy of mass flow rate predictions imposes a significant impact on overall system behaviors, in some scenarios physics-based compressor and expansion valve models are interchangeable with data-driven models when they are easy to implement and computationally affordable. In other words, the proposed dynamic modeling framework enables hybrid modeling that utilizes a mix of data-driven and physics-based component models.

- The GRU heat exchanger model proposed in this work follows a discrete-time dynamical system form with a fixed time step. This is because training data is obtained by sampling simulated system trajectories with a fixed time interval, though the physics-based model for generating the data is solved in a variable-step scheme using solvers implemented in Modelica. By realizing that, the computational speed of GRU models can potentially be further improved for transient simulations if the model adopts a variable-step integration scheme. Therefore,

a way to modify the current model is that the integration time steps taken by Modelica solvers according to system stiffness can be collected in training data and predicted as an output variable. Correspondingly, features and output variables specified in Eq. (15) and (16) are collected at time instances where the solver evaluates system dynamics and determines integration time steps. That means, a training data set can be generated as

$$\begin{bmatrix} \boldsymbol{u}_{t_1} & \boldsymbol{u}_{t_2} & \dots & \boldsymbol{u}_{t_n} \\ \boldsymbol{y}_{t_1} & \boldsymbol{y}_{t_2} & \dots & \boldsymbol{y}_{t_n} \\ \Delta t_{t_1} & \Delta t_{t_2} & \dots & \Delta t_{t_n} \end{bmatrix} \tag{31}$$

where $\Delta t_{t_i}$ is the integration time step at time $t_i$ as one of the model outputs. After a heat exchanger model is trained, the refrigerant charge and coil internal energy can be updated following the same scheme as before, however, with a predicted time step during simulations.

- As stated before concerning simulation speed, the data-driven model is able to run 3.6 times faster than the physics-based model in the case study. However, its speedup potential becomes even more apparent when taking the programming environment into consideration. Since data-driven models are implemented and simulated in Python, comparing the simulation speed with Modelica which compiles models to low-level code and then executes that together with numerical solvers, might not be entirely equitable. Therefore, it can be argued that conducting simulations of data-driven models using an intermediate or low level programming language is anticipated to yield further speedup compared to Modelica. In addition, more computational savings can be expected when employing the proposed modeling approach to large-scale complicated system configurations, especially for applications with complex heat exchanger networks such as VRF systems.

## References

Bendapudi, S., Braun, J.E., Groll, E.A., 2008. A comparison of moving-boundary and finite-volume formulations for transients in centrifugal chillers. International journal of refrigeration 31, 1437–1452.

Bhattacharya, C., Chakrabarty, A., Laughman, C., Qiao, H., 2022. Modeling nonlinear heat exchanger dynamics with convolutional recurrent networks. IFAC-PapersOnLine 55, 99–106.

Chakrabarty, A., Maddalena, E., Qiao, H., Laughman, C., 2021. Scalable bayesian optimization for model calibration: Case study on coupled building and hvac dynamics. Energy and Buildings 253, 111460.

Chen, Z., Fu, X., 2020. Dynamic performance prediction of vehicle variable speed air conditioner based on lstm recurrent neural network. Energy Proceedings 12.

Chen, Z., Xiao, F., Shi, J., Li, A., 2022. Dynamic model development for vehicle air conditioners based on physics-guided deep learning. International Journal of Refrigeration 134, 126–138.

Dong, Y., Qiao, H., Laughman, C., 2024. Physically-constrained hybrid modeling for vapor compression systems, in: 9th Thermal and Fluids Engineering Conference.

Energy, R., 2010. Energy efficiency trends in residential and commercial buildings .

Habtom, R., 1999. Modeling a refrigeration system using recurrent neural networks, in: International Conference on Computational Intelligence, Springer. pp. 47–52.

Hansen, D., Maddix, D.C., Alizadeh, S., Gupta, G., Mahoney, M.W., 2023. Learning physical models that can respect conservation laws. arXiv preprint arXiv:2302.11002 .

Jordan, I.D., Sokół, P.A., Park, I.M., 2021. Gated recurrent units viewed through the lens of continuous time dynamical systems. Frontiers in computational neuroscience 15, 678158.

Kim, D., Ma, J., Braun, J.E., Groll, E.A., 2021. Fuzzy modeling approach for transient vapor compression and expansion cycle simulation. International Journal of Refrigeration 121, 114–125.

Kochenderfer, M.J., Wheeler, T.A., 2019. Algorithms for optimization. Mit Press.

Laughman, C.R., Qiao, H., 2018. On closure relations for dynamic vapor compression cycle models, in: Proceedings of The American Modelica Conference, pp. 9–10.

Li, P., Qiao, H., Li, Y., Seem, J.E., Winkler, J., Li, X., 2014. Recent advances in dynamic modeling of hvac equipment. part 1: Equipment modeling. HVAC&R Research 20, 136–149.

Loka, N.R.B.S., Mejía, N.A., Ortiz, S.T., Gurumurthy, S.K., Monti, A., Rigola, J., Oliet, C., Couckuyt, I., Dhaene, T., 2023. Heat exchanger surrogates for a vapor compression system, in: Modelica Conferences, pp. 599–606.

Ma, J., 2024. Development, Calibration and Validation of A Dynamic Modeling Framework for Air-Source Heat Pumps Under Cycling of Frosting and Reverse-Cycle Defrosting. Ph.D. thesis. Purdue University Graduate School.

Ma, J., Kim, D., Braun, J.E., 2021. Proper orthogonal decomposition for reduced order dynamic modeling of vapor compression systems. International Journal of Refrigeration 132, 145–155.

Qiao, H., Aute, V., Radermacher, R., 2015a. Transient modeling of a flash tank vapor injection heat pump system–part i: Model development. International journal of refrigeration 49, 169–182.

Qiao, H., Laughman, C.R., Aute, V., Radermacher, R., 2016. An advanced switching moving boundary heat exchanger model with pressure drop. International Journal of Refrigeration 65, 154–171.

Qiao, H., Xu, X., Aute, V., Radermacher, R., 2015b. Transient modeling of a flash tank vapor injection heat pump system–part ii: Simulation results and experimental validation. International Journal of Refrigeration 49, 183–194.

Rasmussen, B.P., 2012. Dynamic modeling for vapor compression systems—part i: Literature review. HVAC&R Research 18, 934–955.

Rehmer, A., Kroll, A., 2019. On using gated recurrent units for nonlinear system identification, in: 2019 18th European Control Conference (ECC), IEEE. pp. 2504–2509.

Shao, L.L., Yang, L., Zhao, L.X., Zhang, C.L., 2012. Hybrid steady-state modeling of a residential air-conditioner system using neural network component models. Energy and buildings 50, 189–195.

Wang, J., Lu, X., Adetola, V., Louie, E., 2024. Modeling variable refrigerant flow (vrf) systems in building applications: A comprehensive review. Energy and Buildings , 114128.

Yang, L., Zhao, L.X., Zhang, C.L., Gu, B., 2009. Loss-efficiency model of single and variable-speed compressors using neural networks. international journal of refrigeration 32, 1423–1432.

Yoon, Y.J., Lee, M.H., 2010. Dynamic simulation of vapor-compression cycle using neural networks. International Journal of Control, Automation and Systems 8, 1241–1249.

Yousaf, S., Bradshaw, C.R., Kamalapurkar, R., San, O., 2023. Investigating critical model input features for unitary air conditioning equipment. Energy and Buildings 284, 112823.

Zhang, G., Xiao, H., Zhang, P., Wang, B., Li, X., Shi, W., Cao, Y., 2019. Review on recent developments of variable refrigerant flow systems since 2015. Energy and Buildings 198, 444–466.