Performance Enhancements for Zero-Flow Simulation of Vapor Compression Cycles

Hongtao Qiao[*]   Christopher R. Laughman

Mitsubishi Electric Research Laboratories
Cambridge, MA, USA
`{qiao,laughman}@merl.com`

Abstract

Models that correctly describe the dynamic behavior of vapor compression cycle at low or zero refrigerant mass flow rates are valuable because they can be used to handle low load, on/off cycling and inactive component conditions. However, low- or zero-flow simulation imposes significant computational challenges because of high frequency oscillations in mass flow. We explore techniques that may be used for improving robustness and performance of low- or zero-flow simulation. Comparisons are conducted to demonstrate the efficacy of the proposed techniques. It is shown that these techniques can result in simulations that are more robust and significantly faster than real-time.

Keywords: Modelica, zero flow, heat exchanger modeling, dynamic simulation, vapor compression cycle

1 Introduction

Numerical simulations are widely employed in the modern day HVAC&R (Heating, Ventilation, Air-Conditioning and Refrigeration) industries to assist in the design and optimization of advanced products in response to the increasing pressure of cost reduction and high-energy efficiency standards. With the aid of simulation tools, design engineers can evaluate a conceptual product design on computers instead of building real systems and conducting expensive tests in the laboratory, thereby shortening the time required for design cycles.

In general, vapor compression system simulations can fall into one of two categories: steady-state or transient. The evaluation of the steady-state, full-load performance of vapor compression systems is often used to determine the system capacity and size. However, vapor compression systems rarely operate under steady-state conditions, and dynamic models are more adequate for a realistic representation of the system response. These models are typically used for two types of studies: (1) examining small-scale changes in the refrigerant-side behavior, such as flow instabilities, and (2) examining larger system-level changes, such as system behavior during start-up, shut-down, or defrosting. While many models can typically reproduce small-scale behavior quite accurately, the increased complexity and nonlinearity associated with large-scale transients often results in predictions that have much lower accuracy.

Low and zero-flow phenomena are often encountered in the operation of vapor compression systems with large transients, including on/off cycling of air-conditioning systems, operating mode switch of variable refrigerant flow systems, and reverse-cycle defrosting of heat pump systems as examples. Simulation of system dynamics under such conditions presents numerical challenges, such as problems with robustness and an attendant increase in simulation time due to direction switching flows. In recognition of these problems, Dermont et al. (2016) proposed an approach to improve zero-flow simulation based on a systematic analysis of heat transfer coefficients. Although this approach was shown to increase simulation robustness under (near) zero-flow conditions, the presented use cases ran much slower than real time. In his sole-authored paper, Zimmer (2020) suggested that regularization schemes were required to improve model robustness against zero mass flow without giving further details. Li (2020) discussed the computational improvement from table-based refrigerant property calculation models with Analytical Jacobians. However, the implementation of such methods is a long-term task and requires significant effort. We are thus motivated to explore effective numerical techniques to improve the performance of zero-flow simulations, especially focusing on robustness and improvements in the simulation speed, with a goal of achieving faster than real-time dynamic simulation.

The remainder of the paper is organized as follows. In Section 2, we present a regularized pressure drop model that has significant benefits for these on/off transient simulations. In Section 3, we discuss the advantages and disadvantages of static and dynamic heat transfer coefficient models. In Section 4, we describe the single pressure heat exchanger model and its potential to speed up the zero-flow simulation. Conclusions from this work are then summarized in Section 5.

2 Pressure Drop Model

The finite volume method is often used to discretize the governing equations that describe the dynamics of refrigerant flow because it has been highly successful in approximating the solution of a wide range of thermal-fluid systems and maintaining quantity conservation. In many of these types of models, a staggered grid scheme is utilized to decouple the mass and energy balance equations from the momentum balance equation. As a result, the mass and energy balances are calculated within the volume cells while the momentum balance is calculated within the flow cells, as depicted in Fig. 1.
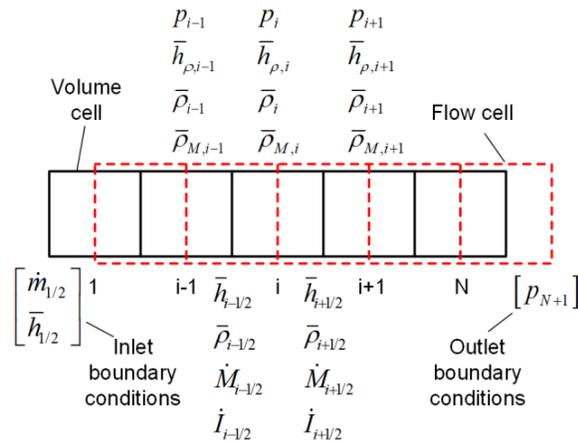
Figure 1. Staggered grid scheme

Since the inertia term, dynamic pressure wave and gravity effect in the momentum equation are usually of minor importance in these applications, they are often neglected in heat transfer analyses to reduce the modeling complexity (Brasz and Koenig, 1983). As a result, the discretized governing equations for 1-D flow are often given as

$$\frac{dM_i}{dt} = \dot{m}_{i-1/2} - \dot{m}_{i+1/2} \tag{1}$$

$$\frac{dU_i}{dt} = \dot{m}_{i-1/2}\bar{h}_{i-1/2} - \dot{m}_{i+1/2}\bar{h}_{i+1/2} + P\Delta z q_i'' \tag{2}$$

$$p_i - p_{i+1} = \Delta p_{f,i} \tag{3}$$

Pressure and specific enthalpy are often selected as state variables to avoid non-linear algebraic equations when they are the independent properties in the media models. The equations of mass and energy can then be rewritten using the time-derivative of pressure and specific enthalpy based on the chain rule.

It is evident from Eq. (3) that the pressure difference between adjacent volumes is solely determined by the frictional pressure loss. The frictional pressure drop is often a complex nonlinear function of Reynolds number and thus mass flow rate, and is often impossible to invert analytically. Since the pressures in each control volume are known at each time step, numerical iterations may be required to solve mass flow rate based on pressure difference, resulting in slower simulations.

Unlike the steady-state models that are often used for system design and require high prediction accuracy, dynamic models are widely used over much wider operating ranges and thus require high robustness and high efficiency, which is sometimes achieved at the expense of accuracy or fidelity to published frictional pressure loss correlations (Idelchik, 1986). Therefore, simplified models are often used to reduce modeling complexity and improve simulation speed. Among those, the following model approximating the frictional pressure loss (Laughman and Qiao, 2018) $\Delta p = f(\dot{m})$ is expressed as

$$\Delta p = K\Delta p_0 \left(\dot{m} / \dot{m}_0 \right)^b \tag{4}$$

where Δp_0 and $\dot{m}_0$ are the parameters in the nominal condition, and K and b (often greater than 1) are curve-fitting constants. This relation is not only less nonlinear than the original correlation-based relations, but it is also easily invertible and can allow the pressure loss to be calculated as a function of the mass flow rate, or vice versa. As such, the resulting system simulations have much faster performance, since the nonlinear dependence on the variety of input variables is removed from the relation and the integrator can take much larger steps. One can easily invert Eq. (4) and obtain the derivative of mass flow rate with respect to pressure drop

$$\frac{d\dot{m}}{d(\Delta p)} = \frac{\dot{m}_0^b}{bK\Delta p_0} \frac{1}{\dot{m}^{b-1}} \tag{5}$$

Eq. (5) suggests that $\dot{m} = f^{-1}(\Delta p)$ is not Lipschitz continuous and becomes increasingly sensitive to pressure drop as it approaches zero and eventually the derivative becomes infinite when mass flow is zero. Meanwhile, we can examine the time derivative of the mass flow

$$\frac{d\dot{m}}{dt} = \frac{d\dot{m}}{d(\Delta p)} \frac{d(\Delta p)}{dt} = \frac{\dot{m}_0^b}{bK\Delta p_0} \frac{d(\Delta p)}{dt} \dot{m}^{1-b} \tag{6}$$

Eq. (6) shows that the time derivative of the mass flow rate is very large when the mass flow rate is small, indicating that Eq. (6) is a stiff ODE. One can make a simple analogy between Eq. (6) and the ODE $y' = -ay \, (a > 0)$. To obtain a stable solution with the Explicit Euler method, the time step must satisfy $\Delta t < 2/a$. When a is very large, the time step becomes very

small, which is exactly the case for Eq. (6) since $a = -\frac{\dot{m}_0^b}{(bK\Delta p_0)}\frac{d(\Delta p)}{dt}\frac{1}{\dot{m}^b}$ is infinite when the mass flow is zero.

The point with infinite derivative is often called singularity point. In consequence of this behavior, the simulation often stalls for off-cycle conditions of vapor compression systems in which mass flow rate is extremely low. A conventional remedy for this behavior is to replace the singular part with locally non-singular substitute resulting in a finite derivative at the point; this process is often referred to as regularization. According to Dermont et al. (2016), a regularized pressure drop correlation is necessary for a complex thermo-fluid model to compute under zero flow conditions. The built-in implementation of power function that employs such regularization for terms in Eq. (4) can be found in `Modelica.Fluid.Utilities.regPow`.

```
function regPow
  extends Modelica.Icons.Function;
  input Real x;
  input Real a;
  input Real delta=0.01;
  output Real y;
algorithm
  y := x*(x*x+delta*delta)^((a-1)/2);
end regPow;
```

This `regPow` function approximates $sign(x)|x|^a$ and is regularized with finite derivatives around the singular point. In this function, the parameter `delta` is used to specify the regularization range. When `abs(x) << delta`, the regularization results in a linear approximation for the original function with the Lipschitz constant to be unity. Although the built-in implementation successfully eliminates the singularity point, it can be further improved. Consequently, one more parameter can be added to `regPow` so that different types of regularization can be formulated.

```
function regPowGen
  extends Modelica.Icons.Function;
  input Real x;
  input Real a;
  input Real delta=0.01;
  input Real b=1;
  output Real y;
algorithm
  y := x^b*(x*x+delta*delta)^((a-b)/2);
end regPowGen;
```

With different values for the parameter `b`, different approximations can be obtained for the original function. When `b = a`, the `regPowGen` function is equivalent to the original power function without regularization. With `b = 1`, the function is same as the built-in function `regPow`. Fig. 2 shows different regularization schemes for the pressure loss relation in the neighborhood around the singularity point. Without regularization, the derivative at the origin is infinite and

the mass flow rate is extremely sensitive to the change in pressure loss around the singularity point, which can cause simulation of low- or zero-flow conditions to crash. The simulation performance does improve with the built-in approach, but is still far from satisfactory because small pressure differences can still result in large variations in mass flow. With `b = 3`, a cubic approximation is used around the singularity point and the Lipschitz constant is much smaller. As a result, the mass flow rate becomes far less sensitive to pressure differences so that the solver can take much larger step sizes.
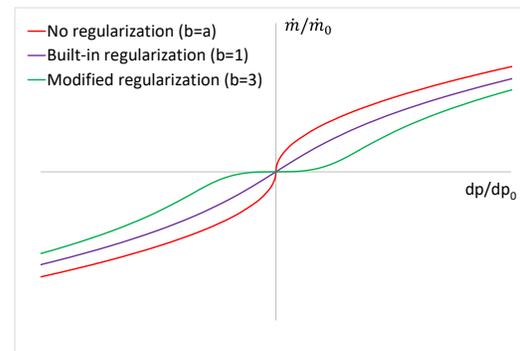
Figure 2. Different types of regularization for pressure loss relation

To evaluate the efficacy of different types of regularization, off-cycle transients of a room air-conditioning system, illustrated in Fig. 3, were simulated. The system ran steadily before it was shut down at 500 sec. The off-cycle period then lasted for 4500 sec and the simulation ended at 5000 sec. As shown in Fig. 4, adequate regularization can make a substantial improvement to the simulation performance. With the built-in `regPow` function for the simplified pressure loss relation, it took more than 23000 sec of CPU time to finish a 5000 sec simulation. However, with modified regularization scheme (`b=3`), it only took around 1700 sec to finish the simulation and CPU time was reduced by more than 10 times. Reducing the sensitivity of the mass flow rate to the pressure difference around the singularity point can thus be a key for faster simulation of low or zero flow conditions. No changes in the component models were required with the proposed regularization scheme. Please note that the proposed cubic approximation should be directly applied to the function that calculates the mass flow rate given the pressure drop, i.e., $\dot{m} = f^{-1}(\Delta p)$. This regularization scheme exhibits no robustness issues in our models since the inverse function $\dot{m} = f^{-1}(\Delta p)$ can be easily computed. If the inverse function $\dot{m} = f^{-1}(\Delta p)$ cannot be obtained, one needs to be cautious when applying the proposed scheme due to the possibly resulting numerical issues.
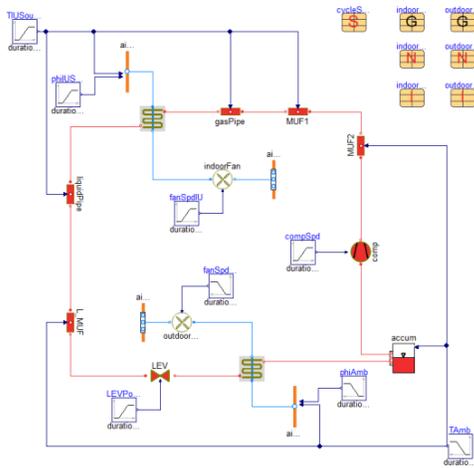
Figure 3. Modelica model of a room air-conditioning system
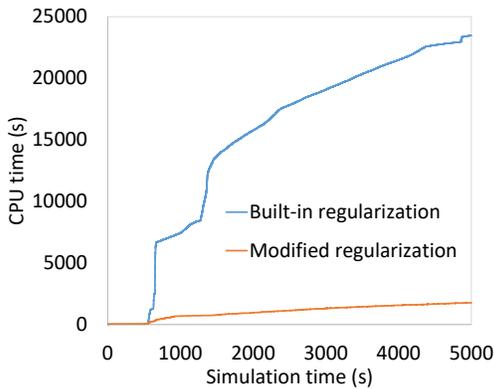
Figure 4. CPU time vs. simulation time with different types of regularization

3 Heat Transfer Coefficient Model

The description of local heat transfer coefficients (HTCs) in the simulation models of thermofluid systems can be particularly challenging, as the correlations are usually formulated with accuracy as the primary concern, and with little regard for computational considerations. Consequently, they can be difficult to incorporate into system-level models of thermofluid systems as they may be extremely nonlinear. Meanwhile, these correlations are usually defined only for specific flow conditions or refrigerant phases, so that there will inevitably be significant discontinuities between regions of the validity for specific correlations. Dynamic simulation presents additional difficulties as the unknown refrigerant mass flow rates, pressures, and specific enthalpies preclude the use of any initial information about the phase of the refrigerant (condensation, evaporation, liquid, or vapor) or the flow regime (laminar or turbulent), so the correlations must

be defined in a manner which encompasses a wide range of flow conditions.

One alternative approach that has been successfully used to mitigate the nonlinearities of detailed heat transfer coefficient correlations has been the creation of simplified models that capture the general trends of those detailed correlations without implementing their complexity. These simplified correlations can be justified via the improved numerical performance of the simulation models, which may not even function with some of the complex correlations found in the literature, as well as the fact that the overall heat transfer coefficient for many refrigerant-to-air heat exchangers is dominated by the air-side heat transfer coefficient, rather than the refrigerant-side heat transfer coefficient.

A wide variety of forms can be used for these relations, depending on the required parametric dependence or level of fidelity to the behavior of the original correlations. For example, we used a simplified heat transfer relation for each phase according to

$$\alpha = K\alpha_0 \left(\dot{m} / \dot{m}_0 \right)^b \tag{7}$$

The constants α_0 for the liquid, two-phase, and vapor flow regions were calculated by coarsely approximating the behavior of the full correlations over their regions of validity, and a trigonometric interpolation method was used to smoothly transition between phases (Richter, 2008).

Laughman and Qiao (2018) proposed the incorporation of dynamics into the closure models to decouple the heat transfer coefficient from the other state variables. This makes the closure variables into state variables of the system, and will decouple the value of the closure variable in the fluid computations with the value of the closure variable calculated from the other state variables. In the case of the heat transfer coefficient, this may be calculated by

$$\frac{d\alpha}{dt} = \frac{1}{\tau}\left(\hat{\alpha} - \alpha \right) \tag{8}$$

where $\hat{\alpha}$ represents the algebraic heat transfer coefficient which can be calculated using either detailed or simplified relations, and α represents the filtered version of the heat transfer coefficient. The parameter τ should be tuned to be substantially faster than other time constants of the system in order to ensure that it will not change the system response.

This dynamic heat transfer coefficient model has proved to be effective at eliminating the spurious oscillations caused by the high gain of $\partial \alpha / \partial x$ in the transition region from vapor phase to two-phase and increase model robustness. However, for the case of off-cycle simulation, the filtered heat transfer model can potentially slow down the simulation because it

increases the number of state variables in the system. As demonstrated in Fig. 5, it took around 3700 sec CPU time to finish the same off-cycle simulation of the air-conditioning system described in Fig. 3 with filtered heat transfer coefficient model, which was 2 times longer than the CPU time of the simulation with static heat transfer coefficient model. It was evident that the filtered model slowed down the simulation remarkably for the first 500 sec after the system was shut down. During this period of time, the refrigerant mass flow rates declined dramatically, resulting in a rapid change in heat transfer coefficients. Setting 'log norm true' during the running simulation when using the Modelica compiler Dymola 2020x (Dymola, 2020) can determine that some of heat transfer coefficient states were causing the integrator to be slow. In summary, the filtered heat transfer coefficient model can help improve model robustness and eliminate the high-frequency numerical oscillations, but not necessarily speed up the off-cycle simulation. It is recommended that modelers try both static and filtered approaches to the heat transfer coefficients to choose a more appropriate approach on a case-by-case basis.
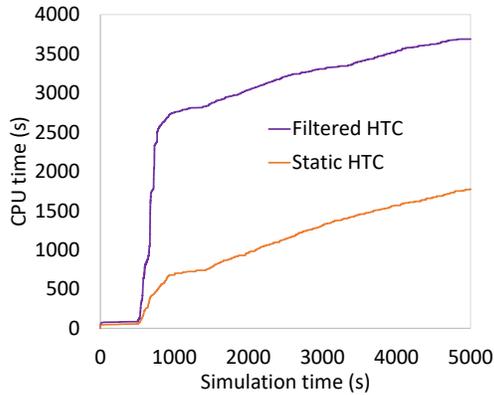
Figure 5. CPU time vs. simulation time with different HTC models

4 Single Pressure Heat Exchanger Model

Heat exchangers usually require particular attention in the modeling of vapor compression cycles because they are the main components where exchange of mass, energy and momentum take place. Accurate mathematical and physical representations for heat transfer and fluid flow phenomena in heat exchangers are always crucial for the overall cycle simulation. In general, three modeling paradigms are often used for heat exchanger simulations. In order of increasing complexity and sophistication, they are the lumped parameter method, the moving boundary method and the finite volume method, respectively.

Lumped parameter models simplify the description of the characteristics of an inherently spatially distributed physical system with mean properties that are assumed homogeneous throughout the heat exchanger by averaging out the spatial variations. With this approach, only the overall mass and energy balances (2 equations) are considered and the thermal behavior of heat exchangers is modeled as a single control volume. Since this approach disregards the spatial variation in properties and the distinct differences of the heat transfer mechanisms between single-phase and two-phase, these models inevitably result in the most inaccurate predictions amongst these three modeling approaches.

Recently, Qiao and Laughman (2022) developed a new low-order heat exchanger model based on the lumped parameter approach. Unlike conventional lumped parameter models, this new model assumes a distribution of refrigerant enthalpy so that the spatial variations of refrigerant properties such as density and specific enthalpy can be taken into account. The overall mass and energy balances to describe the refrigerant dynamics are given as

$$V\left(\frac{\partial \bar{\rho}}{\partial p}\frac{dp}{dt} + \frac{\partial \bar{\rho}}{\partial \bar{h}}\frac{d\bar{h}}{dt}\right) = \dot{m}_{in} - \dot{m}_{out} \tag{9}$$

$$\bar{\rho}V\frac{d\bar{h}}{dt} - V\frac{dp}{dt} =$$

$$\dot{m}_{in}h_{in} - \dot{m}_{out}h_{out} - \left(\dot{m}_{in} - \dot{m}_{out}\right)\bar{h} + \sum_{j=1}^{3} q_{r,j} \tag{10}$$

It is assumed that refrigerant enthalpy varies exponentially in the heat exchanger. Therefore, the local refrigerant quality is determined by

$$x = \frac{x_{in} - x_{out}}{1 - \exp(-1)}\exp(-\zeta) + \frac{x_{out} - x_{in}\exp(-1)}{1 - \exp(-1)} \tag{11}$$

where ζ is the fraction of the heat exchanger covered by the portion from the inlet to the location of interest. Fractions of vapor, two-phase and liquid zone can be readily computed with this enthalpy distribution profile. The mean specific enthalpy of refrigerant in the heat exchanger can also be determined by

$$\bar{h} = \frac{\bar{\rho}_{vap}\bar{h}_{vap}\zeta_{vap}}{\bar{\rho}} + \frac{\left[\bar{\gamma}\rho_g h_g + (1-\bar{\gamma})\rho_f h_f\right]\zeta_{tp}}{\bar{\rho}}$$

$$+ \frac{\bar{\rho}_{liq}\bar{h}_{liq}\zeta_{liq}}{\bar{\rho}} \tag{12}$$

where $\bar{\gamma}$ is the mean void fraction of the two-phase flow. Since $\bar{h}$ is a state variable and is known at each time

step, Eq. (12) can be used to iterate h_{out} so that the entire system is closed. The accuracy of the new models can be significantly improved with the addition of refrigerant enthalpy profile as compared to moving boundary models, but with minimum additional computational cost. A full description of this modeling approach, which is beyond the scope of the present work, can be found in the referenced paper.

In comparison, the moving boundary method is characterized by dividing the heat exchanger into different control volumes, each of which exactly encompasses a particular fluid phase (vapor, two-phase or liquid) and is separated by a moving boundary where the phase transition occurs. In contrast to the distributed parameter models, the number of control volumes in moving boundary models may vary because fluid phases can disappear or appear under large disturbances. These models may consist of at most three control volumes and at least one at a time (6 equations at most). The objective of such models is to capture the thermal behavior inside these control volumes and time-varying position of phase boundaries. Moving boundary models generally result in much faster simulations compared to distributed parameter models due to their small size while more accurately capturing the time-varying dynamics of these systems, but these models are inherently fragile due to their variable model structures. For instance, moving boundary models cannot manage zero or reverse flows because these models are designed with the strict assumption that refrigerant flow enters the heat exchanger from one end, and leaves from the other. These models are either are either over- or underdetermined if these assumptions are violated (Qiao et al., 2016).

Finite volume heat exchanger models are particularly useful for describing spatially dependent phenomena and detailed component performance, such as the effect of local heat transfer and pressure drops or the branching and joining of refrigerant pipes as a result of particular circuiting configurations. As discussed earlier, finite volume models are comprised of an alternative sequence of volume cells and flow cells. The resultant modular nature allows great flexibility in system configurations, and different component models can be seamlessly linked together (Qiao et al., 2015). However, one of the disadvantages of this modeling approach is that it creates many dynamic pressure states. For a model with N control volumes, it has 2N dynamic states, i.e., N pressure states and N specific enthalpy states, resulting in N mass flow rates that need to be computed based on pressure differences. Therefore, 3N equations are needed to solve the model. Under off-cycle conditions, these N mass flow rates will all decline rapidly and each will enter the region where the mass flow rate is highly sensitive to pressure difference. This will inevitably increase the likelihood that the integrator will substantially reduce the step size during the solving process of the model. Based on this reasoning, it is anticipated that the off-cycle simulation can be greatly accelerated if the dependence of mass flow upon pressure difference can be removed. We thus propose a heat exchanger model with a single pressure state and the governing equations are given as

$$A\Delta z \left(\frac{\partial \overline{\rho}_i}{\partial p} \frac{dp}{dt} + \frac{\partial \overline{\rho}_i}{\partial \overline{h}_{\rho,i}} \frac{d\overline{h}_{\rho,i}}{dt} \right) = \dot{m}_{i-1/2} - \dot{m}_{i+1/2} \quad (13)$$

$$A\Delta z \left(\overline{\rho}_i \frac{d\overline{h}_{\rho,i}}{dt} - \frac{dp}{dt} \right) =$$

$$\dot{m}_{i-1/2} \left(\overline{h}_{i-1/2} - \overline{h}_{\rho,i} \right) \overline{h}_{i-1/2} - \dot{m}_{i+1/2} \left(\overline{h}_{i+1/2} - \overline{h}_{\rho,i} \right) + P\Delta z q_i'' \quad (14)$$

In this new modelling approach, the volume cells within the component model share the same pressure. The number of dynamic states is N+1, i.e., one pressure and N specific enthalpies. Mass flow rates between volume cells will be computed through the coupling between the equations of mass and energy (Qiao and Laughman, 2018). The momentum equation is therefore not needed, so that the whole model consists of only 2N equations. It is worthwhile to point out that pressure drop between components is still taken into account in the system model, though the pressure drop is lumped together and calculated at the inlet or the outlet of the component model depending upon the model structure. As a result, the number of pressure states is significantly reduced, while the number of flow models calculating mass flow based on pressure differences is also decreased. These changes can substantially speed up the off-cycle simulation.

With the modified regularization scheme for pressure loss relation, static heat transfer coefficient model, and single pressure heat exchanger model, the same off-cycle simulation finished with around 200s of CPU time, which was 9 times faster than the conventional finite volume models, as shown in Fig. 6. The speedup improvement achieved using all of the techniques discussed in this work was substantial, given that the off-cycle simulation without any of these enhancements was more than 100 times slower. The discrepancies arising from the approximation of lumped pressure drop were minimal, as the system pressures equalize quickly under off-cycle conditions, and pressure drops between volume cells are negligible. Fig. 7 illustrates the suction and discharge pressure transients as well as compressor mass flow during off-cycle. The compressor mass flow instantly dropped to zero after system was shut down, and suction and discharge pressures came to an equilibrium shortly afterwards, which somewhat justified the key assumption of the proposed single pressure heat exchanger modeling approach.

Fig. 8 illustrates a vapor compression system with two evaporators, which was modified based on the results of the single-evaporator system described in Fig. 3. To further demonstrate the efficacy of the proposed enhancement techniques for zero-flow simulation, we present another case study, in which the system described in Fig. 8 was operated normally for the first 500 sec with two active evaporator branches, after which the first evaporator branch was turned off (the fan was off and the valve was closed) and the system continued running before being completely shut off at 3000 sec. The changes in the actuators and the CPU time as a function of simulation time were given in Fig. 9. This simulation finished smoothly and only took 600 sec of CPU time, indicating the effectiveness of the proposed techniques.
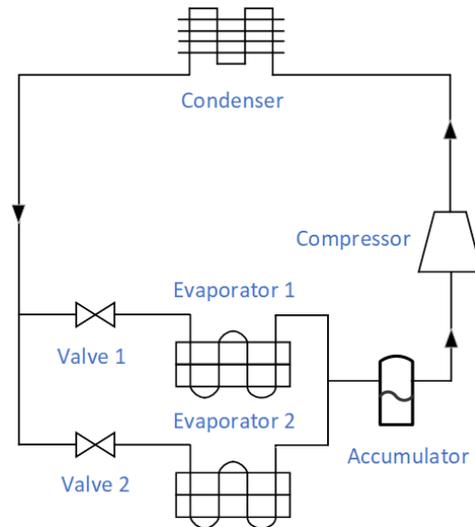
Figure 8. A vapor compression system with two evaporators
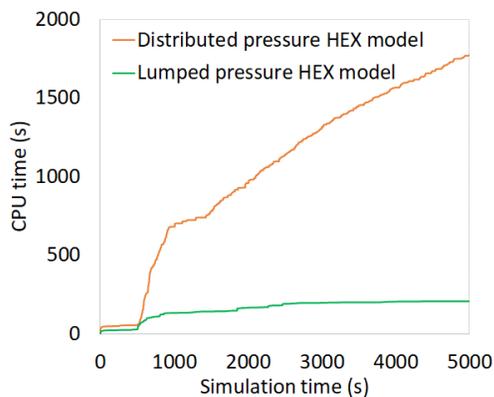
Figure 6. CPU time vs. simulation time with different heat exchanger models
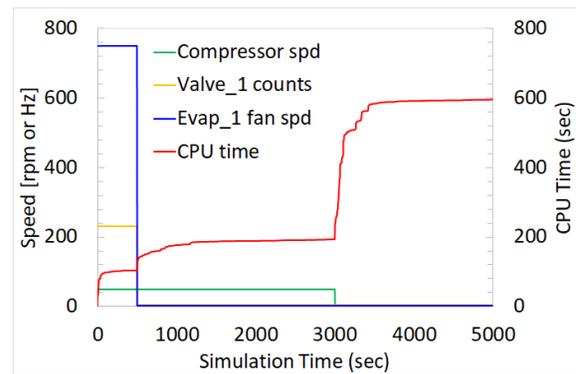
Figure 9. Actuator changes and CPU time vs. simulation time for the system in Fig. 8

5 Conclusions

This paper explored a set of techniques to improve the robustness and speed for zero-flow simulation of vapor compression cycles. It was found that reducing the sensitivity of mass flow to pressure differences was an important key to accelerating the zero-flow simulation. This can be achieved by regularizing the pressure loss relation with cubic approximation in the neighborhood around the singularity point. We also recommend using a static heat transfer model because it reduces the number of dynamic states if no spurious oscillations appear in the simulation. Lumping refrigerant pressure drops at the inlet or outlet of heat exchangers or pipes also demonstrated value in further speeding up the zero-flow simulation. These techniques proved to be efficient to handle refrigerant dynamics in on/off cycling and inactive component conditions.

Figure 7. Pressure and compressor flow transients under off-cycle operation

References

Christopher Laughman and Hongtao Qiao. On closure relations for dynamic vapor compression cycle models. American Modelica Conference, 2018.

Christopher Richter. Proposal of new object-oriented equation-based model libraries for thermodynamic systems. Ph.D. thesis, Technische Universität Braunschweig, Institut für Thermodynamik, 2008.

Dassault Systemes, AB. Dymola 2020x, 2020.

Dirk Zimmer. Robust object-oriented formulation of directed thermofluid stream networks. *Mathematical and Computer Modeling of Dynamical Systems*, Vol. 26, No. 3, pp. 204–233, 2020.

Hongtao Qiao and Christopher Laughman. Comparison of approximate momentum equations in dynamic models of vapor compression systems. In Proceedings of the 16th International Heat Transfer Conference, 2018.

Hongtao Qiao and Christopher Laughman. A Low-Order Model for Nonlinear Dynamics of Heat Exchangers. 18th International Refrigeration and Air Conditioning Conference at Purdue. 2022.

Hongtao Qiao, Christopher Laughman, Vikrant Aute and Reinhard Radermacher. An advanced switching moving boundary heat exchanger model with pressure drop. *International Journal of Refrigeration*, No. 65, pp. 154-171, 2016.

Hongtao Qiao, Vikrant Aute and Reinhard Radermacher. Transient modeling of a flash tank vapor injection Heat Pump System - Part I: Model Development. *International Journal of Refrigeration*, No. 49, pp. 169–182, 2015.

Idelchik, I.E. Handbook of hydraulic resistance. Washington. 1986.

Joost J. Brasz and Kenneth Koenig. Numerical methods for the transient behavior of two-phase flow heat transfer in evaporators and condensers. *Numerical Properties and Methodologies in Heat Transfer*, pp: 461-476, 1983.

Lixiang Li, Jesse Gohl, John Batteh, Christopher Greiner, Kai Wang. Fast simulations of air conditioning systems using spline-based table look-up method (SBTL) with analytical Jacobians. American Modelica Conference, 2020.

Pieter Dermont, Dir Limperich, Johan Windahl, Katrin Prolss and Cartsen Kubler. Advances of zero flow simulation of air conditioning systems using Modelica. Proceedings of the 1st Japanese Modelica Conference, Tokyo, Japan. 2016.