

A Dynamic Analysis of Refrigerant Mass in Vapor Compression Cycles

Bortoff, Scott A.; Deshpande, Vedang M.; Laughman, Christopher R.; Qiao, Hongtao

TR2025-135 September 16, 2025

Abstract

Numerical simulation of a thermofluid vapor compression cycle (VCC) model in Modelica, for example, can exhibit a variation in the total fluid (refrigerant) mass. This paper provides a dynamic analysis of a commonly used VCC model, identifies and analyzes the root cause of this variation, and proposes a number of remedies. The cause lies within the dynamic equations that result from application of the principle of mass conservation. In many common formulations, these equations express the conservation of mass as one or more differential equations that equate the time derivative of mass to zero. The resulting set of n ordinary differential equations (and a number of auxiliary algebraic equations) include the time derivative of a mass constraint function, but not the actual mass constraint function itself. As a result, this modeling formulation has the following properties: (1) equilibrium solutions of the system are neither isolated, nor exponentially stable; (2) a linearization about any equilibrium solution has at least one eigenvalue equal to zero, making an equilibrium solution stable, but not exponentially stable; (3) for a VCC model formulated using two fluid states per control volume, a one-dimensional equilibrium manifold exists containing all of the equilibrium solutions, and is parameterized by the total fluid mass; (4) an $(n-1)$ dimensional, stable, invariant manifold exists transverse to the equilibrium manifold, defined by the mass constraint function, and on which analytic solutions to the model evolve and the total fluid mass remains constant; and (5) numerical solutions may drift off of this manifold, resulting in an observed drift of fluid mass. These properties have consequences for simulation, control design, numerical model reduction, and state estimation. A number of methods to stabilize the mass constraint are proposed and a number of examples that illustrate the behavior, analysis and remedies are provided.

International Modelica Conference 2025

© 2025 MERL. This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

A Dynamic Analysis of Refrigerant Mass in Vapor Compression Cycles

Scott A. Bortoff, Vedang M. Deshpande, Christopher R. Laughman and Hongtao Qiao¹

¹Mitsubishi Electric Research Laboratories, Cambridge, MA, USA {bortoff, deshpane, laughman, qiao}@merl.com

Abstract

Numerical simulation of a thermofluid vapor compression cycle (VCC) model in Modelica, for example, can exhibit a variation in the total fluid (refrigerant) mass. This paper provides a dynamic analysis of a commonly used VCC model, identifies and analyzes the root cause of this variation, and proposes a number of remedies. The cause lies within the dynamic equations that result from application of the principle of mass conservation. In many common formulations, these equations express the conservation of mass as one or more differential equations that equate the time derivative of mass to zero. The resulting set of n ordinary differential equations (and a number of auxiliary algebraic equations) include the time derivative of a mass constraint function, but not the actual mass constraint function itself. As a result, this modeling formulation has the following properties: (1) equilibrium solutions of the system are neither isolated, nor exponentially stable; (2) a linearization about any equilibrium solution has at least one eigenvalue equal to zero, making an equilibrium solution stable, but not exponentially stable; (3) for a VCC model formulated using two fluid states per control volume, a one-dimensional equilibrium manifold exists containing all of the equilibrium solutions, and is parameterized by the total fluid mass; (4) an $(n - 1)$ dimensional, stable, invariant manifold exists transverse to the equilibrium manifold, defined by the mass constraint function, and on which *analytic* solutions to the model evolve and the total fluid mass remains constant; and (5) *numerical* solutions may drift off of this manifold, resulting in an observed drift of fluid mass. These properties have consequences for simulation, control design, numerical model reduction, and state estimation. A number of methods to stabilize the mass constraint are proposed and a number of examples that illustrate the behavior, analysis and remedies are provided.

Keywords: *vapor compression cycle, mass conservation, simulation, dynamic analysis*

1 Introduction

Vapor compression cycles (VCCs) are the primary means of moving heat in almost all modern air conditioning systems. Physically, VCCs consist of one or more hierarchically organized loops of modular components such as heat

exchangers, pipes, valves and compressors. The physics of fluid mechanics, usually including phase change and heat transfer is nonlinear and multi-scale in time and space, resulting in models that consist of large sets of coupled, nonlinear, numerically stiff but sparse differential algebraic equations. As such, the Modelica modeling language has proven to be well suited to this application, with its object oriented structure for organization and tools that support simulation of large sparse DAEs, and has generally found successful application in product development throughout the industry, which is continuously under pressures to innovate.

Often VCC model users need to run simulations over relatively long time horizons. For example, it is important to ascertain the annual performance of a building air conditioner, requiring many simulations over a one year time scale. Standard drive cycles in automotive applications are also relatively long time scale simulations. Several authors have reported that VCC models formulated under standard assumptions (conservation equations) do not conserve refrigerant mass, and that this has adverse practical consequences, especially for long time scale simulations. The refrigerant mass, or charge, seems to drift over the course of a simulation. Some choices of state can help alleviate the problem (Laughman and Qiao, 2015, 2016), although not entirely correct it. Practical methods of resetting states (Anderson et al., 2023) can help to reduce the “refrigerant mass drift,” but require model modification and possibly manual intervention.

These results address the symptom of refrigerant mass drift in numerical simulations, but not its root cause. In this paper, we analyze a common model formulation of a VCC, identify mathematically the root cause of refrigerant mass drift, and describe two methods that can remedy the situation. The dynamic analysis results in an improved understanding of the structure and behavior of the resulting system of DAEs, with some perhaps surprising consequences and implications. For example, we show that equilibrium solutions to the systems of DAEs are stable, but not exponentially stable, which is often assumed to be the case. In fact, there is an equilibrium manifold of stable solutions to the set of DAEs. The equilibrium manifold is parameterized by the total refrigerant mass, and unavoidable numerical errors incurred during a numerical simulation will result in mass drift along the manifold.

This has implications for simulation, model-based control system design, and state estimation.

Fortunately, the structure suggests several remedies, which may be implemented as modifications to standard modeling constructs and result in exponentially stable models that do not exhibit refrigerant mass variation. This paper is organized as follows. We begin Section 2 with a VCC dynamic model resulting from application of standard mass, energy and momentum conservation assumptions. We analyze this model geometrically, proving existence of an integral submanifold of dimension one less than the state dimension, and existence of a one-dimensional equilibrium manifold. We show how this structure implies drift specifically in the refrigerant mass in a numerical simulation, illustrated by a simple VCC example. We then present two remedies in Section 3, with simulation examples. In Section 4 we also show how this behavior can manifest in a mechanical system model, which aids in understanding the situation, and shows that the results are general in nature and not restricted to only thermofluid cycles. We close in Section 5 by summarizing the results and stating a conjecture regarding more complex VCCs.

2 VCC Dynamics

A dynamic model of the thermofluid physics of a VCC generally begins from mathematical statements of distributed, one-dimensional mass, energy and momentum balance, such as

$$\frac{\partial(\rho A)}{\partial t} + \frac{\partial(\rho A v)}{\partial x} = 0 \quad (1a)$$

$$\frac{\partial(\rho v A)}{\partial t} + \frac{\partial(\rho v^2 A)}{\partial x} = -A \frac{\partial P}{\partial x} - F_f \quad (1b)$$

$$\frac{\partial(\rho u A)}{\partial t} + \frac{\partial(\rho v h A)}{\partial x} = v A \frac{\partial P}{\partial x} + v F_f + \frac{\partial Q}{\partial x}, \quad (1c)$$

where P is a pressure, v is a velocity, u is a specific internal energy, ρ is a density, Q is a heat transfer rate, A is a cross-sectional area, V is a volume, F_f is a frictional pressure drop, x is a scalar spatial coordinate and t is time. These may be discretized in the spatial coordinate x resulting in a set of ordinary differential equations (ODEs)

$$\frac{d(\rho_j V_j)}{dt} = \dot{m}_k - \dot{m}_{k+1} \quad (2a)$$

$$\begin{aligned} \frac{d(\dot{m}_i l)}{dt} &= \rho_j v_j^2 A_j - \rho_{j+1} v_{j+1}^2 A_{j+1} \\ &+ \frac{A_j + A_{j+1}}{2} (P_{j+1} - P_j) + F_{f,i} \end{aligned} \quad (2b)$$

$$\begin{aligned} \frac{d(\rho_j u_j A_j)}{dt} &= \dot{H}_k - \dot{H}_{k+1} \\ &+ v_j j A_j (P_{j+1} - P_j) + v F_{f,i} + \dot{Q}_j, \end{aligned} \quad (2c)$$

where the indices j and k correspond to gridding methods, and \dot{m}_k denotes the mass flow rate at the boundaries of a finite control volume. See (Pat, 1980; Tummescheit, 2002;

Laughman and Qiao, 2015, 2016) for details of modeling procedures and Modelica realizations.

State selection is dependent on the particulars of an application, but generally two states are selected for each control volume: A mixture variable, and another variable. Herein we choose pressure P and specific enthalpy h , which are common choices for VCC applications. However, the main results are not dependent on this specific choice. For compactness, define the full state $x = [P_1, h_1, P_2, h_2, \dots, P_m, h_m] \in \mathbb{R}^n$ where m is the number of control volumes and $n = 2m$, and express (2) as

$$\dot{x} = f(x, u), \quad (3)$$

where $u \in \mathbb{R}^p$ is a vector of control inputs, such as a compressor speed or an expansion valve opening. For simplicity, this formulation ignores states resulting from consideration of heat transfer to metal. These would have no bearing on the main results. Note that a Modelica model may not explicitly compute f , but rather it may express the dynamics in an implicit form. However, f exists mathematically and will make use of this fact.

In (2) the total refrigerant mass M is an explicit function of state:

$$M = g(x) := \sum_{i=1}^m \rho_i(x_{2i}, x_{2i+1}) V_i. \quad (4)$$

Note that neither g nor M appear in (2). However, (2a) is a statement that $\dot{g} = 0$. In typical use, g is used to set the initial condition $x(t_0) = x_0$ to correspond to a total refrigerant mass M_0 using

$$M_0 = g(x_0). \quad (5)$$

2.1 Main Results

Since $\dot{g} = 0$, we have

$$\dot{g} = \frac{\partial g}{\partial x} \cdot \dot{x} = \frac{\partial g}{\partial x} \cdot f = 0. \quad (6)$$

Therefore, at any equilibrium solution x_e satisfying $f(x_e, u_e) = 0$, we have

$$\left. \frac{\partial g}{\partial x} \right|_{x=x_e, u=u_e} \cdot f(x_e, u_e) = 0. \quad (7)$$

Expanding f into a Taylor's series about (x_e, u_e) , we have

$$\left. \frac{\partial g}{\partial x} \right|_{x=x_e, u=u_e} \cdot A = 0, \quad (8)$$

where A is the Jacobian of f with respect to x , i.e., $A = \left. \frac{\partial f}{\partial x} \right|_{x=x_e, u=u_e}$. Therefore

$$U_0 := \left. \frac{\partial g}{\partial x} \right|_{x=x_e, u=u_e} \quad (9)$$

is the left eigenvector corresponding to a zero eigenvalue of A . Define V_0 to be the right eigenvector of A corresponding to the zero eigenvalue of A . Since A has one

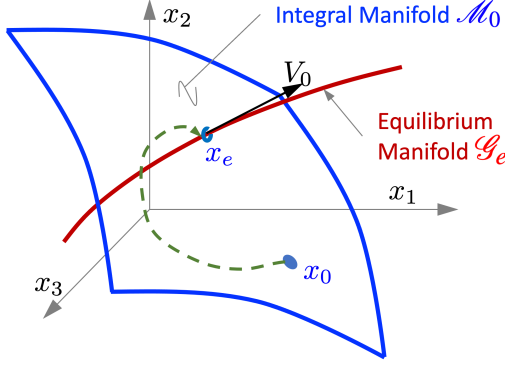


Figure 1. Integral manifold \mathcal{M}_0 , equilibrium manifold \mathcal{G}_e , and a solution trajectory (green) for initial condition $x_0 \in \mathcal{M}_0$, converging to the stable equilibrium $x_e \in \mathcal{M}_0 \cap \mathcal{G}_e$.

zero eigenvalue (and the remaining $n - 1$ eigenvalues are assumed to lie in the open left half plane), the equilibrium solution x_e is stable, but not exponentially stable. Further, x_e is not isolated, since $\tilde{x}_e = x_e \pm \varepsilon \cdot V_0$ is also an equilibrium for any $\varepsilon > 0$, by definition of the eigenvalue and eigenvector. Therefore, the linearized system

$$\delta \dot{x} = A \delta x + B \delta u$$

has an equilibrium subspace defined by the span of V_0 , and therefore the nonlinear system (2) has a one-dimensional equilibrium manifold \mathcal{G}_e , which is tangent to V_0 at x_e , defined by those states x_e satisfying

$$0 = f(x_e, u_e) \quad (10a)$$

$$0 = g(x_e) - M_e \quad (10b)$$

which is parameterized by the refrigerant mass M_e (and also u_e). Finally, as a consequence, the nonlinear system has an $n - 1$ dimensional integral manifold \mathcal{M}_0 , implied by the fact that f lies within the tangent space of the $n - 1$ dimensional submanifold defined by $dg = 0$, where d is the gradient operator. (An *integral manifold* \mathcal{M}_0 is invariant to the dynamics, meaning $x_0 \in \mathcal{M}_0 \rightarrow x(t) \in \mathcal{M}_0$ for all $t \geq t_0$.)

The situation is diagrammed in Figure 1, where an initial condition x_0 is defined to satisfy (10), the analytic solution $x(t)$ evolves on the $n - 1$ dimensional integral manifold \mathcal{M}_0 , which contains x_0 , converging to the equilibrium solution x_e which lies in the intersection of \mathcal{M}_0 and \mathcal{G}_e . The equilibrium x_e is not exponentially stable because the eigenvector V_0 corresponding to the zero eigenvalue is tangent to \mathcal{G}_e . As such, there is a family of $n - 1$ dimensional integral manifolds \mathcal{M}_k , all transverse to \mathcal{G}_e , defined by the initial condition x_0 . (These foliations are not shown in Figure 1 for clarity.) All of these results follow from the fact that A has one eigenvalue equal to zero, basic linear algebra and elementary nonlinear differential equation theory found in reference texts such as (Vidyasagar, 1993), (Khalil, 2002) and (Isidori, 1989).

2.2 Some Implications

This geometric structure has implications for numerical simulation, control design, model reduction and state estimation applications. Analytically, a solution $x(t)$ evolves on the $n - 1$ dimensional integral manifold \mathcal{M}_0 , and the effective dimension of the state vector is $n - 1$, given an initial condition x_0 , which determines the refrigerant mass M_0 . However, a numerical simulation evolves in the full n -dimensional state space. Inevitable numerical errors at each discrete time step of any solver algorithm will result in an approximate solution that is not invariant to \mathcal{M}_0 , i.e., the solution will drift off \mathcal{M}_0 , onto some other \mathcal{M}_k . This drift is in exactly the direction of varying the refrigerant mass. Further, since the dynamics in the \mathcal{G}_e direction are not exponentially stable, these errors are not corrected step to step. In effect, the drift is like a one dimensional random walk. The constraint $0 = g(x) - M_0$ is not stabilized.

In fact, there is a hidden state in the dynamics, corresponding to the direction of dg and locally associated to the eigenvector V_0 . In a spectrally decomposed coordinate system this state is an open loop integrator driven by “noise” generated by numerical simulation errors, step to step. (The random walk analogy is very close, except the VCC simulation is deterministic.) For some simulations, this “noise” may be biased, which may be caused by simulation errors near the saturation curve, where the change in density is extreme, and in this case, a numerical simulation may seem to leak its entire charge of refrigerant.

It has been stated by some authors that the “mass is not a state” in such models, but this is not entirely correct. It is not an explicit element of x . However, the refrigerant mass is a state of the system, appearing as an open loop integrator in the spectrally decomposed coordinates. Since the x coordinates are used for simulation, the mass state is hidden, mixed among all the other states. This implies that any time domain simulation will exhibit drift in the refrigerant mass, regardless of the coordinates used, and regardless of the integration algorithm used. Some choice of coordinates (Laughman and Qiao, 2015, 2016) and also measures taken to enforce the mass constraint (Anderson et al., 2023) may result in a reduced magnitude of drift, owing to improved error control, but do not cause the mass constraint to be enforced.

For control, it is common to make use of model linearizations. Care must be taken in reducing the order of such models. Hankel norm methods which remove modes that are both weakly controllable and weakly observable, work only for exponentially stable systems. Matlab provides convenient tools for such purpose, but these will first extract the unstable (and weakly stable) part of the system by spectral decomposition, reduce the stable part, and then put the unstable part back, assuming that is what the user wants! Of course, for these applications, the user needs to remove the mode corresponding to the zero eigenvector first, because this mode is a modeling artifact. The true state dimension is $n - 1 = 2m - 1$.

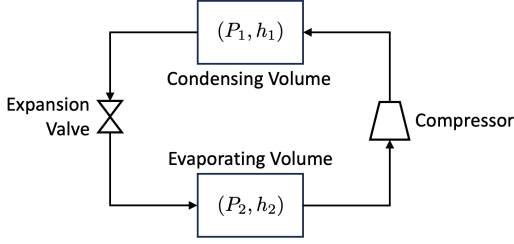


Figure 2. Two-volume vapor compression cycle.

2.3 An Example

Consider the simple two-volume VCC diagrammed in Figure 2, consisting of two control volumes labeled “condensing” and “evaporating” each with pressure-enthalpy states (P_1, h_1) and (P_2, h_2) , respectively, a compressor and an expansion valve. The latter are modeled as algebraic functions relating the inlet and outlet states. Here the control input $u = [u_1, u_2]$ with the compressor speed u_1 and expansion valve count u_2 . The state vector is $x = [P_1, h_1, P_2, h_2]$, and the dynamics are

$$\dot{x} = f(x, u), \quad x(t_0) = x_0 \quad (11a)$$

$$M = g(x) = \rho_1(P_1, h_1)V_1 + \rho_2(P_2, h_2)V_2. \quad (11b)$$

Linearizing the dynamics about an equilibrium solution confirms that 3 eigenvalues are in the open left half plane, and one is at the origin, with its left eigenvector U_0 aligned with the gradient of g . A simulation result with initial condition $P_1 = 2.6 \text{ MPa}$, $h_1 = 420 \text{ kJ/kg}$, $P_2 = 1 \text{ MPa}$ and $h_2 = 230 \text{ kJ/kg}$, constant values of the compressor speed (30Hz) and expansion valve (100 counts), and constant boundary conditions results in the transient response plotted in Figure 3. The refrigerant mass drifts for the 15s, coming to rest after the transient has transpired. Depending on the application, this amount of drift may or may not be acceptable. Importantly, it is entirely driven by numerical solver error, more or less the result of integrating a random signal.

2.4 VCC Models in (P, h, ρ) Coordinates

Formulating the VCC dynamics in (P, h, ρ) coordinates has been suggested as an approach to mitigate mass drift (Laughman and Qiao, 2015, 2016). These coordinates were derived by first expressing the dynamics in (P, ρ) -coordinates, and then adding a dynamic equation for h . Following this approach, assume the change of coordinates from (P, h) to (P, ρ) is well-defined in a region of interest. In the (P, ρ) -coordinates, denote the state vector as $y = [P_1, \dots, P_m, \rho_1, \dots, \rho_m]^T$. All the results from Section 2.1 also carry over to these coordinates. Next consider a linearized model around any equilibrium solution $\bar{y} \in \mathbb{R}^{2m}$, ignoring for simplicity the effect of any inputs,

$$\delta \dot{y} = A \delta y + B \delta z \quad (12a)$$

$$\delta \dot{z} = C \delta y, \quad (12b)$$

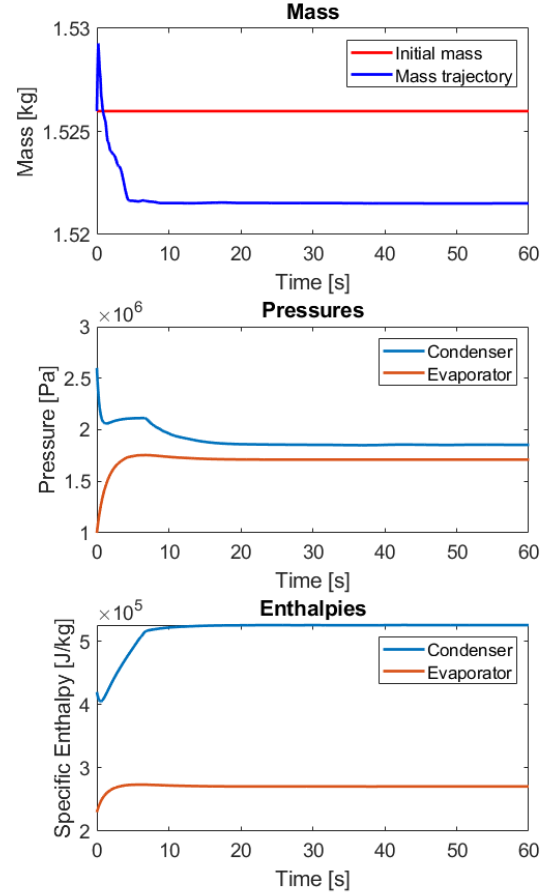


Figure 3. Refrigerant mass drift during initial transient.

where $\delta y = y - \bar{y}$, and $z \in \mathbb{R}^m$ is the enthalpy vector h , $\bar{z} \in \mathbb{R}^m$ is the equilibrium enthalpy vector, and $\delta z = z - \bar{z}$. The linearized enthalpy term δz is explicitly pulled out and expressed as an output function because (P, h, ρ) -coordinates use it as a third state in each control volume. Note that the linearized dynamics in (P, ρ) coordinates are $\delta \dot{y} = (A + BC) \delta y$.

In this notation, the corresponding linearization in (P, h, ρ) coordinates is

$$\delta \dot{y} = A \delta y + B \delta z \quad (13a)$$

$$\delta \dot{z} = C \delta y = CA \delta y + CB \delta z. \quad (13b)$$

Note that the δz in (13b) is used on the right hand side of (13a), which means that in the original (P, h, ρ) -coordinate formulation, the h states, computed as solutions to m ODEs, are used in the $2m$ ODEs defining \dot{P} and $\dot{\rho}$. Rewriting (13) in matrix form

$$\begin{bmatrix} \delta \dot{y} \\ \delta \dot{z} \end{bmatrix} = \begin{bmatrix} A & B \\ CA & CB \end{bmatrix} \begin{bmatrix} \delta y \\ \delta z \end{bmatrix}, \quad (14)$$

makes it clear that the system has m eigenvalues at zero, since the last m rows are linearly dependent on the first $2m$. As such a system expressed in (P, h, ρ) -coordinates will have a linearization with $m + 1$ eigenvalues at the origin,

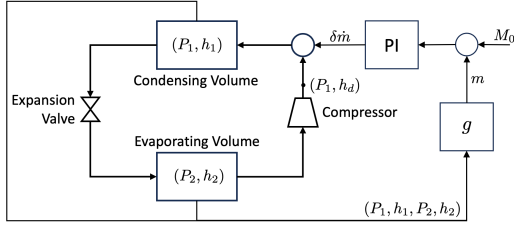


Figure 4. Two-volume vapor compression cycle.

and the nonlinear system will have an equilibrium manifold of dimension $m + 1$ parameterized by the mass, and also the error between the m values of enthalpy h computed by the additional m differential equations and the algebraic constraint $h = h(P, \rho)$ that was differentiated. The improved accuracy in mass conservation reported in (Laughman and Qiao, 2015, 2016) is likely due to better solver error control in the density dynamics, but drift in a higher-dimensional manifold resulting in non-physical results is possible in long duration simulations.

3 Drifting Mass Remedies

We present two remedies. First, we can exponentially stabilize the total amount of mass in a cycle using a virtual feedback loop with integral action. Consider again the simple two-volume VCC diagrammed in Figure 2, now with a virtual mass feedback loop which acts to regulate the total amount of mass to the constant M_0 , shown in Figure 4. This method requires minimal modification of the original model to allow for an additional virtual control input. The virtual input might be an added mass flow rate signal at the discharge port of the compressor, as shown in Figure 4, or alternatively it might be an added enthalpy signal at the discharge port of the compressor.

The PI feedback exponentially stabilizes the total mass, which can be understood by the Root Locus diagrammed in Figure 5. The “plant” here is the system from the virtual input to the measured mass output, which is $m = g(x)$. The plant includes a single pole at the origin, as discussed in Section 2, and $n - 1$ poles in the open left half plane, where $n = 4$ for this example. The feedback loop including the PI compensator adds a second pole at the origin, and a zero in the open left half plane, at $s = -0.2$ for this example. Closing the feedback loop with positive gain causes the double poles at the origin to break away and move into the open left half plane, exponentially stabilizing the equilibrium. The action of feedback may change the dynamics of (3) but only slightly, because the dynamics on the integral manifold is invariant and the dynamics on the integral manifold are relatively insensitive to the virtual mass input. This is clear on Figure 5, where the 3 stable poles move very slightly from open to closed loop (the closed loop location is plotted over the open loop in the figure). The integral manifold \mathcal{M}_0 is not changed by the PI feedback.

Figure 6 shows the result of a simulation in which the

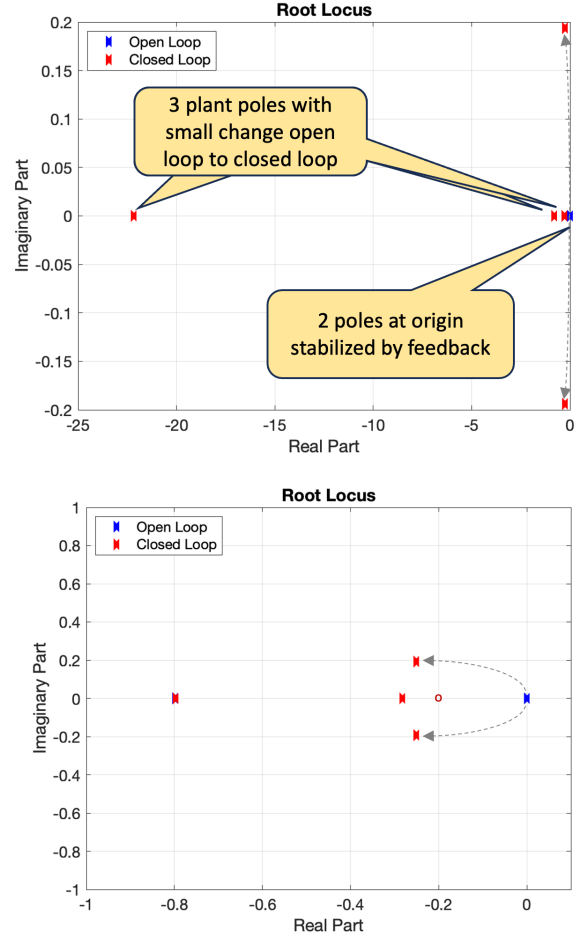


Figure 5. Root locus for the 2-volume system with virtual mass PI feedback (top), magnified to show poles and zeros near the origin (bottom). The PI zero is placed at $s = -0.2$. The 3 stable plant poles are relatively insensitive to the feedback, without visible displacement in the figure. The two unstable poles at the origin are moved to complex conjugate locations by the action of the virtual feedback loop.

virtual mass feedback is active, for the same initial conditions as in Figure 3. The feedback actively stabilizes the total mass and ensures that the equilibrium solution x_e is isolated and exponentially stable.

3.1 Stabilizing the constraint

Alternatively, following Gear (Gear, 1988; Brenan et al., 1996), a second method to stabilize the mass constraint is as follows. The modeled system (3) does not include the refrigerant mass constraint $0 = g(x) - M_0$. But Gear shows that the augmented system

$$\dot{x} = f(x, u) + \mu \cdot dg(x) \quad (15a)$$

$$0 = g(x) - M_0, \quad (15b)$$

where

$$dg = \frac{\partial g}{\partial x} \quad (16)$$

and μ is scalar Lagrange multiplier state, enforces the mass constraint explicitly and has an identical analytic so-

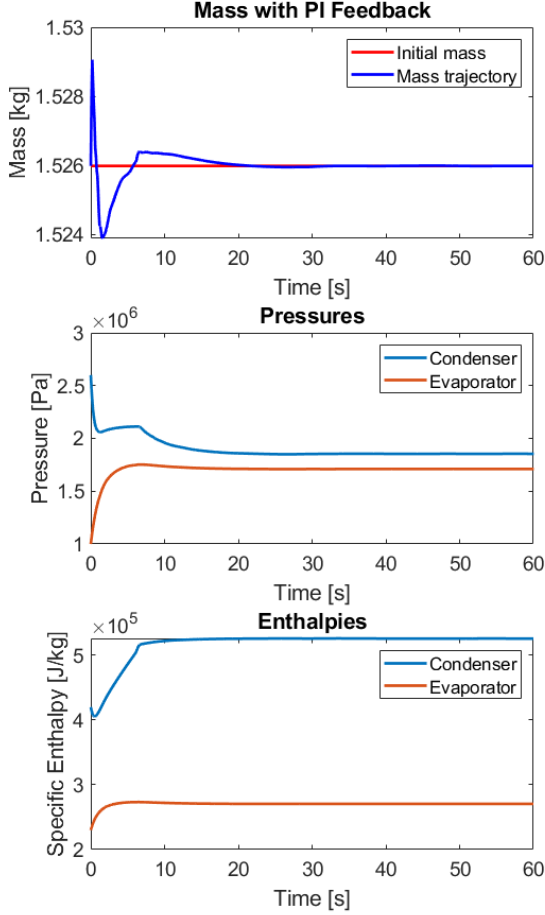


Figure 6. Refrigerant mass under the action of virtual PI feedback control for same initial condition as Figure 3.

lution as (3) with $\mu = 0$. Numerical solutions of (15) will therefore not suffer from drift from \mathcal{M}_0 . However, the index of (15) is 2, so this system cannot be simulated directly using standard solvers.

The index can be reduced using the available methods provided in Modelica tools such as Dymola, but this could result in a complex and highly coupled model equations, since the gradient term dg contains all of the states and as a practical matter, most VCC models (3) are large in dimension to begin with. Alternatively, we may reduce the index manually using Baumgarte’s method (Baumgarte, 1983). This involves replacing the constraint equation (15b) with a linear combination of g and \dot{g} , so that the new constraint equation is Hurwitz (roots in the open left half plane),

$$\dot{x} = f(x, u) + \mu \cdot dg(x) \quad (17a)$$

$$0 = \dot{g}(x) + \alpha_0 \cdot (g(x) - M_0), \quad (17b)$$

where the constant $\alpha_0 > 0$. System (17) is index 1 and may be simulated directly using standard solver algorithms. Note that \dot{g} in (17b) is $dg \cdot (f + \mu \cdot dg)$ which not the same as \dot{g} in Section 2, because the definitions of x and \dot{x} , respectively, have changed by the addition of the $\mu \cdot dg$ term in (17a).

To implement this scheme, the original model (3) needs modification by adding a single equation (17b) and adding

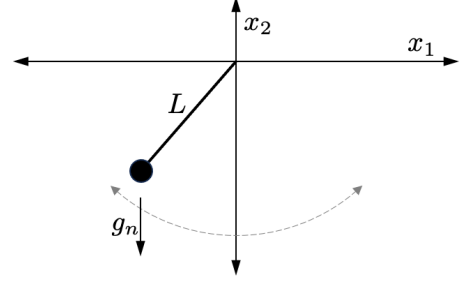


Figure 7. Cartesian pendulum.

the output injection terms $\mu \cdot dg$ to each of the state equations. This would require modification to the base classes of the modeling package, since they were probably not intended for this purpose, but the changes are relatively minor. System (17) may be less computationally complex compared to that obtained by reducing the index of (15) from 2 to 1 using the standard method of substitution of variables (or “dummy derivatives”). The method increases the dimension of the set of equations from n to $n + 1$. But as a result, the constraint is stabilized, and the resulting equilibrium solution x_e is unique, isolated and exponentially stable. Numerical solutions will exhibit a small amount of error in the refrigerant mass during a simulation (as do any of the other states), on the order of the solver tolerance, but will not drift far from the initial condition because of the stabilization.

Implementing this approach for VCC models requires modifications to the base classes, which remains part of future work. Nevertheless, we illustrate the method using a simpler example in the following section.

4 Toy Pendulum Example

To show these results are not restricted to only VCC models, the dynamic analysis and remedies can be illustrated using the quintessential Modelica example: The Cartesian pendulum, diagrammed in Figure 7, modeled as

$$\dot{x}_1 = v_1 \quad (18a)$$

$$\dot{x}_2 = v_2 \quad (18b)$$

$$\dot{v}_1 = -\lambda \cdot x_1 - b \cdot v_1 \quad (18c)$$

$$\dot{v}_2 = -\lambda \cdot x_2 - b \cdot v_2 - g_n \quad (18d)$$

$$0 = g(x_1, x_2) = x_1^2 + x_2^2 - L^2, \quad (18e)$$

where (x_1, x_2) is the Cartesian location of the bob, (v_1, v_2) is its velocity, λ is the Lagrange multiplier, g_n is the acceleration due to gravity, $b = 0.1$ is a damping coefficient introduced to stabilize the pendant position, and length $L = 1$. Realizing this model in Modelica, compiling in Dymola and simulating for $x_1(0) = 0.1$ gives the results in Figure 8, where Dymola reduces the index (from 3) and dynamically selects two states, here x_1 and v_1 . There is of course no drift because the dimension of the dynamic states equals the number of degrees of freedom.

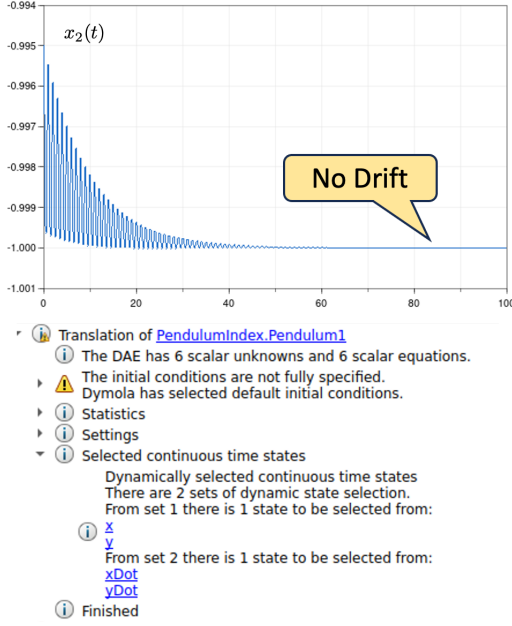


Figure 8. State selection results and simulation trajectory $x_2(t)$ of model (18).

Now consider the system with the constraint (18e) replaced with its derivative,

$$\dot{x}_1 = v_1 \quad (19a)$$

$$\dot{x}_2 = v_2 \quad (19b)$$

$$\dot{v}_1 = -\lambda \cdot x_1 - b \cdot v_1 \quad (19c)$$

$$\dot{v}_2 = -\lambda \cdot x_2 - b \cdot v_2 - g_n \quad (19d)$$

$$0 = \dot{g}(x_1, x_2) = 2x_1v_1 + 2x_2v_2. \quad (19e)$$

This is analogous to VCC model with (19e) being similar to the mass flow rate constraint (2a). Realizing this model in Modelica, compiling in Dymola and simulating for $x_1(0) = 0.1$ gives the results in Figure 9, where Dymola reduces the index (from 2) and dynamically selects *three* states, here x_1 , v_1 and x_2 . Linearizing at the pendant equilibrium solution and computing eigenvalues results in the two stable eigenvalues that we expect, and a third eigenvalue at zero, associated with an eigenvector in the x_2 -direction, precisely the direction of the missing constraint $g(x_1, x_2) = 0$ at the pendant equilibrium $(0, -1)$. The numerical solution is observed to drift along the vertical x_2 direction in Figure 9, away from the point $(0, -1)$.

Finally we augment (19) using Gear's method (Gear, 1988). The position constraint $0 = g(x_1, x_2)$ is added to the set, and the Lagrange multiplier μ multiplied by the gradient of g is added to the right-hand side of the ODEs (19a)-(19d), resulting in an index-2 DAE. Then we manually reduce the index from two to one using Baumgarte's method (Baumgarte, 1983), replacing the constraint equation $g = 0$ with $\dot{g} + \alpha_0 \cdot g = 0$, where $\alpha_0 > 0$, ($\alpha_0 = 1$ for

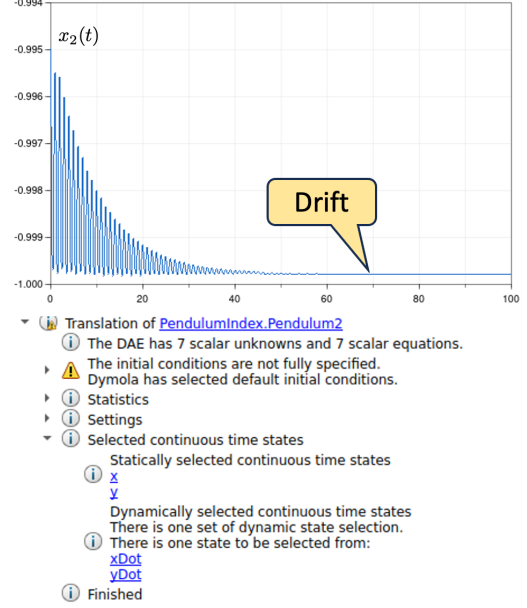


Figure 9. State selection results and simulation trajectory of $x_2(t)$ of model (19).

convenience) resulting in the index-1 DAE

$$\dot{x}_1 = v_1 + \mu \cdot 2x_1 \quad (20a)$$

$$\dot{x}_2 = v_2 + \mu \cdot 2x_2 \quad (20b)$$

$$\dot{v}_1 = -\lambda \cdot x_1 - b \cdot v_1 \quad (20c)$$

$$\dot{v}_2 = -\lambda \cdot x_2 - b \cdot v_2 - g_n \quad (20d)$$

$$0 = 2x_1v_1 + 2x_2v_2 \quad (20e)$$

$$0 = \dot{g} + \alpha_0 \cdot (x_1^2 + x_2^2 - L^2). \quad (20f)$$

Again, there is a subtle but important point to consider: The \dot{g} in (20f) is not the same as the \dot{g} in (19e), because \dot{x} has changed from the second to the third example. Realizing this in Modelica, compiling in Dymola and simulating for $x_1(0) = 0.1$ gives the results shown in Figure 10. Interestingly, Dymola reduces the index to 0, and dynamically selects three states, here x_1 , v_1 and x_2 , the same states selected for the second case. Linearizing the system at the equilibrium solution results in the two stable eigenvalues associated with the pendulum that we expect, and a third exponentially stable eigenvalue at $-\alpha_0$, associated with the stabilized constraint equation $\dot{g} + \alpha_0 \cdot g = 0$. The equilibrium solution is now exponentially stable. Although a numerical simulation of this system will suffer from a small amount of numerical error in the constraint ($g = 0$ is not exactly enforced), caused by Baumgarte's method (Bortoff, 2019) this can be controlled through the solver tolerance, and importantly the solution will not exhibit constraint drift, as shown in Figure 10.

5 Conclusion

Numerical simulations of some dynamic models of vapor compression cycles based on standard conservation equations may not conserve refrigerant mass, resulting in sim-

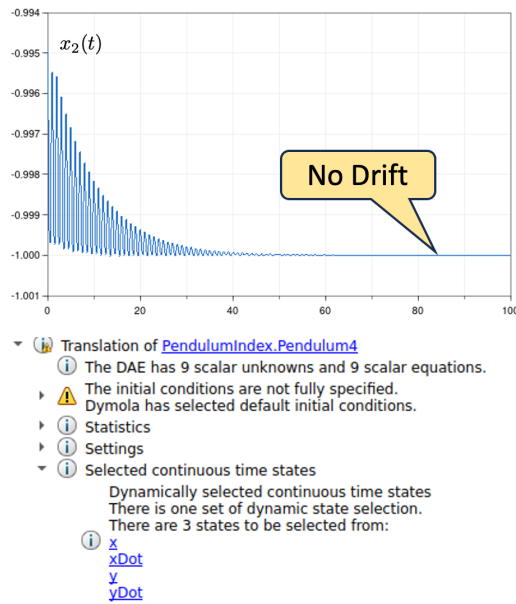


Figure 10. State selection results and simulation trajectory of $x_2(t)$ of model (20).

ulation failures especially for relatively long simulation times. This paper provides a dynamic analysis to ascertain mathematically the root cause of so-called refrigerant mass drift, resulting in an improved understanding of the dynamic model structure and behavior. The key result is that models derived using conventional assumptions and using conventional coordinates (two fluid states per control volume) are not exponentially stable, as is often assumed, and the n -dimensional state space may be decomposed into an $n - 1$ dimensional integral manifold, and a transverse 1-dimensional equilibrium manifold. Numerical solutions will inevitably drift in the direction of the equilibrium manifold, manifesting as a varying refrigerant mass. Models that make use of higher dimensions such as (P, h, p) for each control volume, will exhibit higher dimensional equilibrium manifolds that may result in numerical drift in these spaces, leading to unphysical results. Two methods to remedy are proposed and shown to be effective with a thermofluid VCC example and also a classic pendulum example.

References

- Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing Co., 1980.
- Daniel Anderson, John Batteh, Matthis Thorade, and Lixiang Li. Mass conservation in vapor compression cycles: A method for ensuring consistency with redundant dynamic states. In *Proceedings of the 15th International Modelica Conference*, pages 693–702, 2023.
- J. W. Baumgarte. Stabilization of constraints and integrals of motion in dynamic systems. *Computer Methods in Applied Mechanics and Engineering*, 1:1–16, 1972.
- J. W. Baumgarte. A new method of stabilization for holonomic constraints. *ASME Journal of Applied Mechanics*, 50:869–870, 1983.
- Scott A. Bortoff. Using Baumgarte’s method for index reduction in Modelica. In *Proceedings of the 13th International Modelica Conference*, pages 333–342, March 2019.
- K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM, 1996.
- C. W. Gear. Differential-algebraic equation index transformation. *SIAM Journal on Scientific and Statistical Computing*, 9(1), 1988.
- Alberto Isidori. *Nonlinear Control Systems*. Springer-Verlag, 1989.
- Hassan K. Khalil. *Nonlinear Systems: Third Edition*. Prentice Hall, 2002.
- Chris R. Laughman and Hongtao Qiao. On the influence of state selection on mass conservation in dynamic vapor compression cycle models. *Mathematical and Computer Modeling of Dynamical Systems*, 23(1):262–283, Dec. 2016.
- Christopher R. Laughman and Hongtao Qiao. Mass conserving models of vapor compression cycles. In *Proceedings of the 11th International Modelica Conference*, pages 759–767, Versailles, France, 2015.
- Hubertus Tummescheit. *Design and Implementation of Object-Oriented Model Libraries using Modelica*. PhD thesis, Lund Institute of Technology, 2002.
- M. Vidyasagar. *Nonlinear Systems Analysis: Second Edition*. Prentice-Hall, 1993.