Proceedings
of the 4th International Modelica Conference,
Hamburg, March 7-8, 2005,
Gerhard Schmitz (editor)

M. Tiller
Ford Motor Company, USA
Development of a Simplified Transmission Hydraulics Library based on Modelica.Fluid
pp. 237-243

Development of a Simplified Transmission Hydraulics Library based on Modelica.Fluid

Michael M. Tiller
Research and Advanced Engineering, Ford Motor Company
mtiller@ford.com

Abstract

Modeling of hydraulic systems often leads to systems of equations that are stiff and difficult to solve. In many cases, stiffness of these systems can be traced to orifices and relatively small volumes within the model. Frequently, such volumes and orifices are only present to facilitate explicit state-space formulations of the underlying conservation principles.

In an effort to create more efficient models and to eliminate the need for insignificant or non-physical contributions from such components, the new `Modelica.Fluid` library [1] introduces a structured set of base classes (leveraging new features in the Modelica language) from which fluid component models can be built. These base classes allow for a wider range of component configurations by eliminating the need for extraneous volumes and orifices in hydraulic schematics.

Using the `Modelica.Fluid` library as a foundation, another library has been developed that includes hydraulic components for hydraulic transmission modeling. The models are aimed at addressing lingering performance and robustness issues with hydraulic circuits in transmission models and include several useful simplifications. Because these component models use a first-principles formulation (*i.e.* conservation of mass and energy), it is possible to mix simplified or idealized components with models that include complex dynamics. As such, model developers can focus on the dynamics of interest (*e.g.* dynamics associated with the design of a specific spool valve in a transmission) while still capturing the basic functional behavior for the other components in the system. The result of this approach is a practical continuum between functional and predictive modeling.

Keywords: Transmission, VMA, hydraulics, DAEs

1 Goals

Models for transmission hydraulics usually have one of two purposes. The first purpose is to be a *functional* reproduction of an existing or proposed transmission design. In this case, the response of the model is only intended to reproduce the functional behavior of the actual transmission hydraulics. Such models would naturally include delays, approximate rates of response, etc.

The other purpose is to be a predictive model of a particular transmission design. Such a model is referred to as a design-oriented model because it can be used to conduct "what if?" studies on potential design candidates. The key requirement for this kind of model is that it should not only be sensitive to the relationship (both transient and steady-state) between the inputs to the model and the outputs but it should also properly predict the hydraulic response *as a result of changes to the design parameters* (orifice sizes, volumes, diameters, *etc*). This latter possibility requires considerable physical detail and a solid, first-principles understanding of component behavior.

The goal of the component library described in this paper is to provide a path to move between these two types of representations easily. In this way, a single model of the hydraulic system can be developed that can be selectively refined to serve both purposes. In some cases, it is useful to consider both purposes in the same model. For example, as part of the design process for a particular valve a design-oriented model of that valve can be used in conjunction with functional models of the remaining components. This not only brings the potential for faster simulations to speed up the design process but also integrates nicely the cascading nature of system engineering based development processes.

2 Functional vs. Predictive

2.1 Background

The simplest way to demonstrate the differences between the formulation and performance of functional and predictive models is to look at an example in detail. However, even basic hydraulic circuits are a little difficult to understand without some background. For this reason, we will first explain how a basic hydraulic regulator valve works before moving on to a complete circuit model. For simplicity, we consider the single solenoid value, shown in Figure 1.

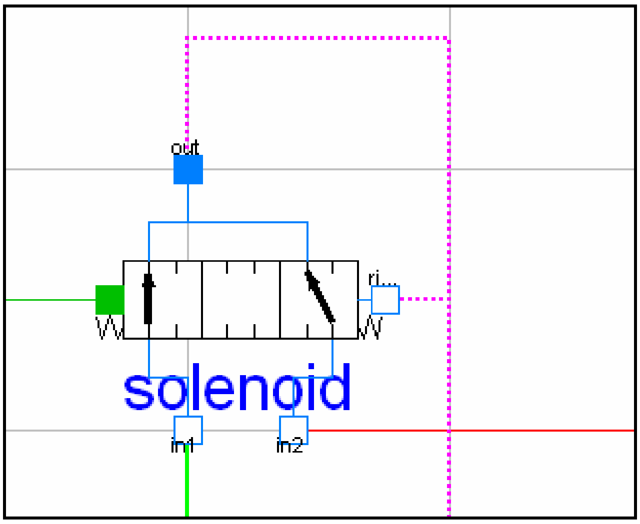

Figure 1: Sample Regulator Valve

A common characteristic of a regulator valve is that an axial force is applied to both sides of the valve. In most cases, the force is the result of a pressure applied by a fluid over the exposed area on that side of the valve. But the force can be generated by other means as well (*e.g.* in the case of a solenoid valve, an electro-magnetic force is applied to one side). The "output" port of the valve (shown in Figure 1, at the top of the valve) can be fed from two potential sources (indicated in Figure 1 by the ports on the bottom of the valve) depending on the position of the valve. In the case of all valves shown in this paper, the larger the force on the left side of the valve, the more flow will come from the bottom right input port. Conversely, the larger the force on the right side of the valve, the more flow will come from the bottom left input port. In simple terms, the output pressure will be biased toward the pressure in the input port on the opposite side of the larger force. To tune the performance of the valve, an "offset" force can be generated using a preloaded spring inside the valve body.

2.2 Functional Model

The model shown in Figure 2 represents an example of a pressure control circuit for an automatic transmission.

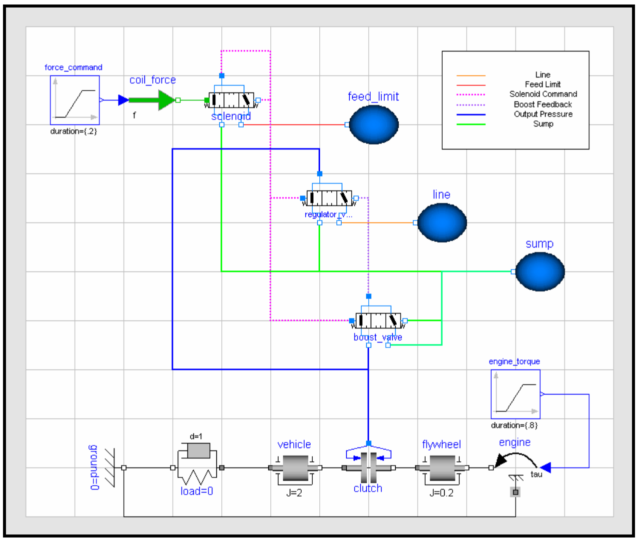

Figure 2: Sample Pressure Control Circuit

Using the background provided in Section 2.1, we can now explain the circuit shown in Figure 2 and deduce the following functional behavior:

- A force command signal (upper left) is sent to an electric coil in the solenoid. This coil applies the commanded force to the left side of the solenoid valve.
- Because the output pressure of the solenoid is also the pressure applied on the right side, the solenoid valve will seek a position that balances the electric coil force with the "output pressure" of the valve. In this way (and with a gain that depends on the areas involved), the output pressure of the solenoid valve is controlled.
- The solenoid output pressure is applied to the left sides of both the regulator and the boost valve. The boost valve (bottom valve) is designed with a preload such that it does not open until a critical pressure has been reached. Above that critical pressure, its output pressure starts dropping to the sump pressure. Otherwise, its output pressure follows the regulator valve pressure.
- Before the critical pressure of the boost valve is reached, the regulator valve functions much like the solenoid valve because its output pressure is effectively the balancing force on the right side. Again, given the areas involved a certain gain is achieved.

However, once the boost valve starts to open, the pressure on the right side of the regulator quickly drops giving the pressure on the left a greater mechanical advantage (and thus increasing the gain significantly).

- Finally, the regulator valve output is applied to the clutch. The clutch plates do not come into contact with each other until a critical pressure is reached. Prior to this, the flow into the clutch fills the gap that forms as the plates moves. This filling effect results in a delay between the regulator output pressure and the applied clutch pressure.

2.3 Detailed Model

The functional description in Section 2.2 describes how the circuit is supposed to function. However, the functional description assumes a steady-state response with no dynamic effects. In reality, there are many dynamic effects.

For example, each valve includes a small volume on each end that fills and empties as the force balance changes. In addition, the flow is regulated by orifices which open and close as the valve moves. The behavior of the orifices is non-linear and very sensitive to the machining of the spool itself. Furthermore, these circuits are designed to provide large flow rates which means the fluid itself can build up a significant amount of momentum. Finally, the compressibility of the transmission fluid (transmission fluid often includes a significant amount of trapped gas) combined with the small mass of the spool can result in high-frequency oscillatory responses.

While the dynamics described in this section are on a much smaller time scale than the functional dynamics described previously, they can have a very significant effect on noise and vibration in the mechanical system. As such, these kinds of detailed models (and the staggering amount of geometric component data they require) are very useful in the design of the pressure control circuits.

2.4 Structural Differences

The biggest difference between the functional and predictive models is the presence of dynamic terms. For example, the momentum balance on the regulator valve spool can be expressed as[1]:

$$\sum F = p_L A_L - p_R A_R - m\ddot{x} - kx = 0$$

[1] For simplicity, so-called “flow forces” (*i.e.* reaction forces from changing the fluid momentum) are neglected.

where p_L is the pressure on the left side, A_L is the area on the left side, p_R is the pressure on the right side, A_R is the area on the right side, m is the mass of the spool and x is the position of the spool.

Similarly, the mass balance for the volume on the left side of the valve can be expressed as:

$$Q_i - Q_o - \dot{\rho} A(x - x_0) - \rho A\dot{x} = 0$$

where Q_i is the mass flow rate into the volume, Q_o is the mass flow rate out of the volume, ρ is the density, A is the cross-sectional area and x is the position of the valve.

The inclusion of the capacitive elements makes the formulation of the problem simpler because most, if not all, of the equations can be written as explicit differential equations, *e.g.*,

$$\dot{x} = v$$

$$\dot{v} = \frac{p_L A_L - p_R A_R - kx}{m}$$

$$\dot{\rho} = \frac{Q_i - Q_o + \rho A v}{A(x - x_0)}$$

However, as we will discuss later the simpler formulation is actually much more expensive to solve because of the high frequency dynamics in the system. Because many of these capacitive elements contribute nothing to the overall functional behavior, one simplification is to eliminate them. Rather than deleting them from the model, they can be “logically” deleted by setting their capacitances (*e.g.* m and A) to zero. As we can see from the differential equations, if we continue to rely on the explicit differential equations, such an approach would lead to a singular system of equations because the denominators would go to zero. However, if we reconsider the structure of the problem given that these capacitive terms are zero and allow purely algebraic constraints to appear in the problem formulation (effectively turning the problem into a system of differential-algebraic equations), then we get the relatively simple system:

$$x = \frac{p_L A_L - p_R A_R}{k}$$

$$Q_i = 0$$

While many traditional dynamic system analysis tools are based on the notion that the dynamics *must* be characterized in terms of ordinary differential equations (ODEs), Modelica specifically broadens the general problem definition to support differential-algebraic equations (DAEs) [2]. This broader problem definition combined with sufficient symbolic manipulation [3] (as in Dymola, [4]) means that such simplifications are a practical means of formulating simplified systems of equations using exactly the same component models.

The equations in this section touch on only a few of the structural differences between functional and predictive models. Modelica also provides features for expressing more complex behavior like parametric behavior formulations (often used to describe diodes or clutches [5]) and piecewise linear expressions (used to describe convected property balances in `Modelica.Fluid`). All of this means that many of the limitations that exist when forced to cast component behavior in terms of explicit differential equations can be completely eliminated. As a result, alternative formulations that are more natural, flexible and computationally efficient, like the ones used for the components in this paper, become possible.

It is worth noting that useful behavioral descriptions are typically sufficiently complex that the resulting algebraic equations are both non-linear and coupled. As a result, after symbolic manipulation simultaneous non-linear systems of algebraic equations (*i.e.* "algebraic loops") often emerge from the explicit differential equations. However, the cost of the non-linear iterations is often much less than the cost of resolving the fine details associated with higher-order dynamics.

3 Comparisons

In this section we will quantify many of the differences in structure and performance between these two types of models. We start with the functional model and then include the higher-order dynamic effects. In addition to comparing the simulation results, we will also consider how these effects change the overall structure of the problem and what impact this has on simulation time.

3.1 Functional Validation

Before we compare the structure and performance of these two types of models, we should first perform a basic validation of the model. Note that while the example in this paper was chosen to provide a "real-world" context to the issues, the model itself was created specifically for this paper and is not a validated automatic transmission circuit.

There are two main characteristics of interest in this circuit. The first is to confirm the effect of the boost valve. This should create a "knee" in the pressure response of the circuit. The other effect is the filling of the clutch volume as the clutch is stroked.

Figure 3 shows the functional validation of the circuit by plotting regulator valve output pressure and clutch pressure as a function of coil force. When the coil force reaches 35 Newtons, we can see the knee in the output pressure. In addition, the difference between the clutch pressure and the regulator valve output pressure is caused by the need to fill the gap formed as the clutch is moved into position.

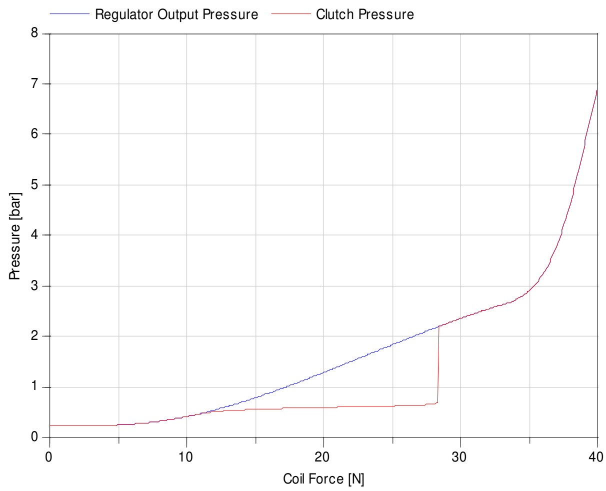

Figure 3: Functional Validation

3.2 Adding Predictive Details

To demonstrate the impact of including predictive details in the model, we will modify the functional model to include 3 important physical effects: spool dynamics, valve orifices and valve volumes. These are just a few typical examples of complexities that are required to truly predict the response of the hydraulic systems. Other effects are also significant (*e.g.* fluid inertia, mechanical limits, compressible media) but for simplicity they will be neglected.

For the functional model, the spool mass is assumed to be zero. As a result, the position of the spool in the functional model is determined by the steady state force balance on the spool. However, in the case where the spool has non-zero mass, the balance of the axial forces determines the acceleration of the spool[2].

To simplify the calculation of the valve output pressure, the functional model prescribes the output pressure by blending the two input pressures continuously as a function of the spool position. The appropriate mass flow rates to achieve this are computed implicitly. This is not particularly physical

[2] In addition to the spool mass, some damping must be introduced as well.

because it assumes that the valve can flow any amount of mass. In reality, such flow rates are limited by the sizes of the various orifices. For the predictive model, the flow through each path in the valve is computed explicitly based on the pressure drops between the ports. While the equilibrium position of the valve will be identical in each case, the presence of orifices results in constraints on how quickly the control circuit can respond. In addition, the nature of the orifice equation typically results in some numerical issues.

The final detail is the filling and mixing associated with the volumes at the ends of the valves. In the functional model, the volumes at the ends of the valves are neglected (*i.e.* no mass or energy capacitance). For the detailed model not only is this volume included, but is varies with spool position.

While these dynamics complicate the response of the circuit, they do not change the overall functional behavior. Figure 4 shows the response of the circuit with these physical details included. The conditions are nearly identical to those used to generate Figure 3.

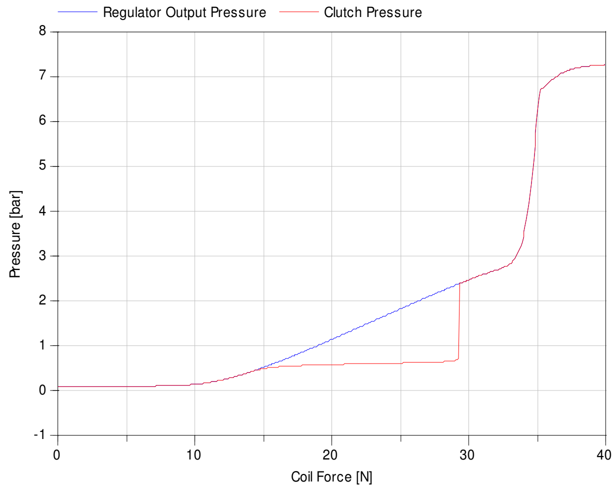

Figure 4: Detailed Model Validation

It should be noted that the conditions chosen for validation in Figure 4 are such that the dynamics have little impact. The biggest feature is the "smoothing" of the edges. However, because the progression of coil force proceeded on a quasi-steady time scale, the dynamics are not visible. The main purpose of Figure 4 is simply to show that with the physical details included, the circuit is still functionally equivalent.

3.3 Transient Response

To highlight the implications that the various physical details have on the transient response, we need to drive the circuit under more realistic operating conditions. Figure 5 shows what a typical solenoid command might look like. The square pulse at the start is used to fill the clutch volume. At the end of the square pulse, the friction materials in the clutch should just be coming into contact. At approximately 0.4 seconds, the solenoid force drops to allow the clutch to engage smoothly. The force is then slowly ramped up to increase the capacity of the clutch. Once the clutch is locked, the coil force jumps up to keep the clutch firmly engaged.

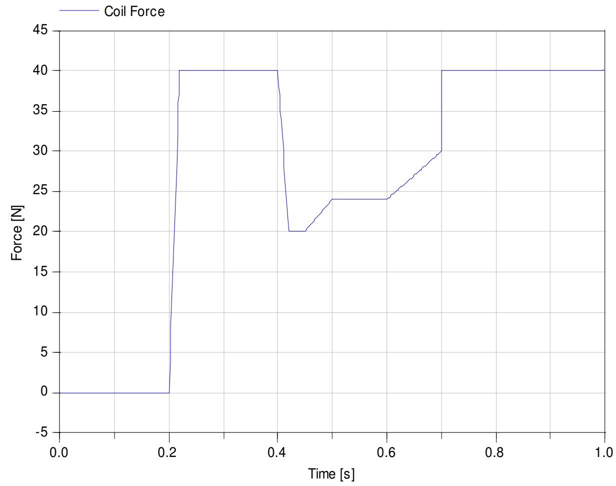

Figure 5: Solenoid Force Command

The functional model response is shown in Figure 6. Note that the clutch pressure does not respond until after the clutch volume is filled. Once the volume is filled, the response of the clutch pressure closely follows the force command profile shown in Figure 5.

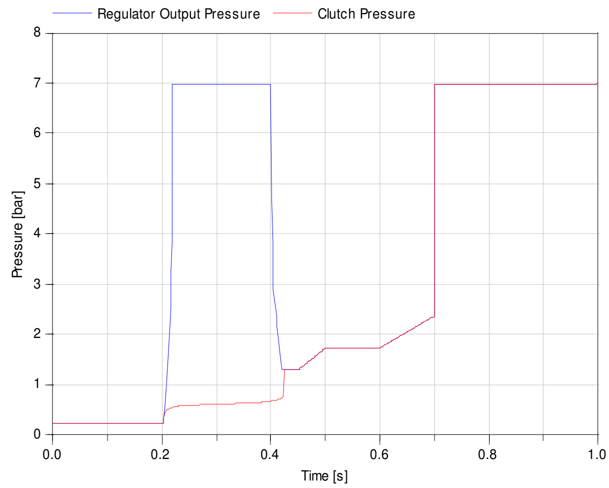

Figure 6: Functional Model Response

By comparison, Figure 7 shows how the more detailed model responds. Rather than responding cleanly to the coil force command, the regulator valve output pressure fluctuates as the spool settles into a quiescent state. In particular, the spool oscillates significantly in response to step changes in the coil force. Most of these oscillations occur before the clutch starts to engage, but some of them can clearly be seen in the clutch pressure response. Another interesting effect shown in Figure 7 is the delay in the engagement of the clutch. This is due to the fact that the regulator output pressure does not drop immediately to a pressure that is in proportion to the coil force. Instead, the clutch pressure drops lower than it did in the functional model which causes the clutch volume to briefly empty before recovering. These dynamics introduce an additional delay before the friction materials come in contact.

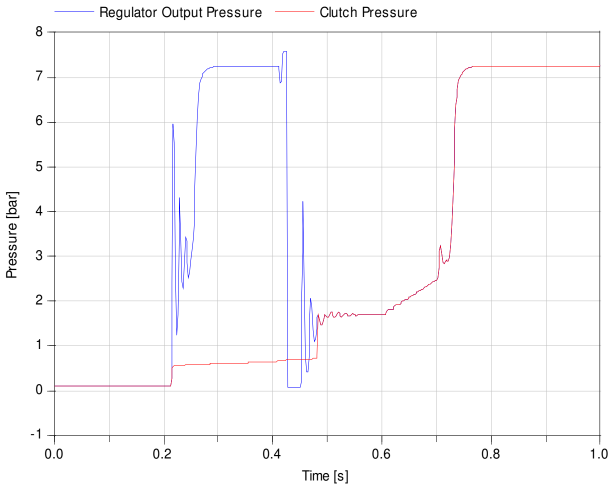

Figure 7: Detailed Model Response

3.4 Structure and Performance

Table 1 highlights some of the structural and performance differences between the simulations shown in Figure 6 and Figure 7. Because the functional model is approximately 5 times faster than the predictive models, it is useful for proving out control strategies or for performing hardware-in-the-loop testing. The performance differences between these types of models could become even more significant as additional physical details are added or as the overall complexity of the circuit increases.

	Functional Model	Detailed Model
# of states	6	20
Linear system sizes	{6, 2, 11}	{6, 10, 4, 13}
Nonlinear system sizes	{8, 9, 6}	{3, 13}
CPU time	0.09 [s]	0.46 [s]

Table 1: Quantitative Comparison

Another way to visualize the differences in the dynamics of the two models it to visualize the poles in each model. In this way, the range of time constants and natural frequencies can be quickly assessed. Figure 8 shows a map of the two poles present in the functional model. One of the poles has a time constant of 1 millisecond and corresponds to a specific first order response introduced in the filling model. The other pole has a natural frequency of 0.08 Hz and corresponds to the mechanical response of the clutch-inertia system shown at the bottom of Figure 2.

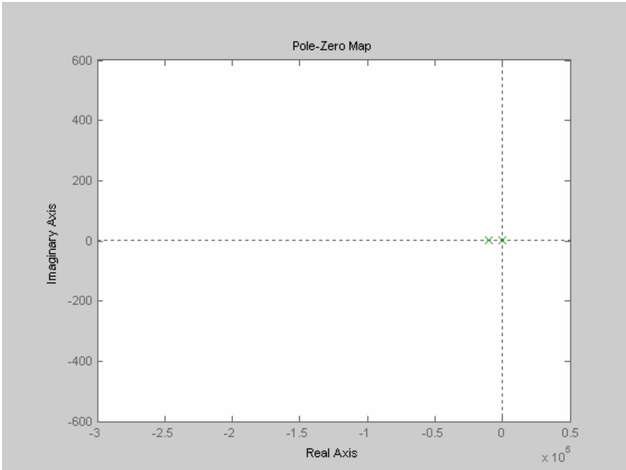

Figure 8: Poles in the Functional Model

For comparison, the poles of the predictive model are shown in Figure 9. An important difference between the poles shown in this figure and the ones shown in Figure 8 is that the dynamics in the predictive model are non-linear. As a result, the positions of the poles vary as a function of the states in the predictive model. For this reason, Figure 9 overlays the values of the poles (computed via linearization) at various times during the predictive model response shown in Figure 7 to demonstrate the range of the dynamics.

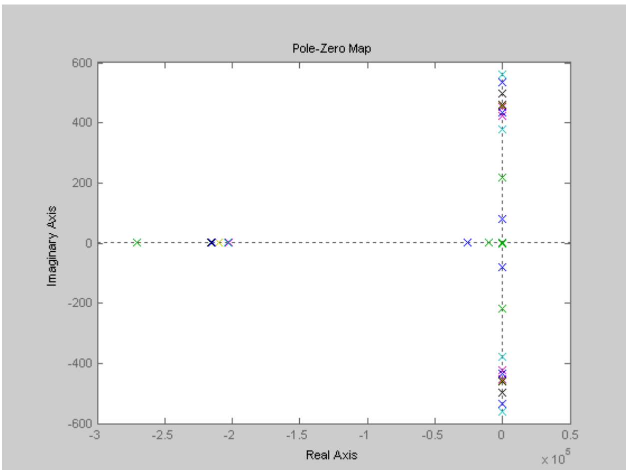

Figure 9: Poles in the Predictive Model

The lowest natural frequency shown in Figure 9 has a value of 18.7 Hz and it exists prior to the first change in solenoid pressure. The highest natural frequency in the system is 90.7 Hz and it appears during the drop in solenoid pressure at approximately 0.41 seconds.

A more intangible quality to these models is the underlying robustness. While Table 1 compares the performance of simulations that were run using these two models, what it does not show is the fact that the detailed model is less robust numerically than the functional model. Singularities associated with vanishing volumes, ill-posed Jacobians, *etc*. can not only have an even greater detrimental impact on the simulation time, they can prevent the simulation from completing at all.

4 Conclusions

A common issue in modeling applications is including the appropriate level of detail for the task at hand. During the initial design phase of a circuit like the one shown in Figure 2, it is important to quickly verify the functional performance of the circuit and/or the control strategy behind it. Simple models can quickly confirm the steady state clutch pressure achieved for a given solenoid force. Then, as the design process focuses on finer details (orifice diameters, spool masses, *etc*) additional geometric information can be added that allows additional dynamics to be considered.

This paper highlights several advantages of using Modelica for hydraulic system modeling. The first advantage is the ability to leverage the `Modelica.Fluid` and `Modelica.Media` libraries. Careful thought has gone into the formulation of these libraries to leverage as much of the potential of the Modelica language as possible while still providing a relatively straightforward framework for developing components. These libraries can now serve as the foundation for the development and exchange of hydraulic component models.

The other advantage of using Modelica for hydraulic systems is the ability to express idealizations that fall outside the typical formulations. It is no longer necessary to build models from alternating "flow-volume" pairs or to consider only behavioral models that lead to explicit differential equations. Instead, Modelica allows the expression of a broader class of behavioral models which, through symbolic manipulation, can be simplified down to relatively simple and efficient algebraic relationships.

Acknowledgments

The central idea for this paper came after reviewing various models used within Ford to model transmission hydraulics. As such, I would like to thank Chad Griffin and James McCallum for their willingness to share and discuss their models. I would also like to thank James for the time he graciously spent explaining many of the nuances of hydraulic control circuits. Greg Pietron, Bill Tobler and James McCallum also provided valuable feedback while reviewing this paper. Finally, I would also like to thank Bill Tobler and Steve Frait for taking the time to discuss and explain the behavior of specific components discussed in this paper.

References

[1] Elmqvist H., Tummescheit H., and Otter M., "Object-Oriented Modeling of Thermo-Fluid Systems", *Proceedings of the 3rd International Modelica Conference*, Linköping, Sweden, 2003.

[2] *"Modelica – A Unified Object-Oriented Language for Physical Systems Modeling"*, Version 2.1, January 30th, 2004. Modelica Association, http://www.modelica.org/documents/ModelicaSpec21.pdf

[3] Mattsson, S. E. and Söderlind G., *"Index Reduction in Differential-Algebraic Equations Using Dummy Derivatives"*, SIAM J. Sci., Comput., 14(3):677-692, May 1993.

[4] Brück D., Elmqvist H., Olsson H., Mattsson S.E., "Dymola for Multi-Engineering Modeling and Simulation", *Proceedings of the 2nd International Modelica Conference*, Oberpfaffenhofen, Germany, 2002.

[5] Tiller M. M., "Introduction to Physical Modeling with Modelica", *Kluwer Academic Publishers*, 2001. ISBN 0-7923-7367-7.