Avoiding Ill-Conditioning on Object-Oriented Models of Hydraulic Actuation Systems

Andrea Giorgio Bartolini [*] **Francesco Casella** [**]
Alberto Leva [**]

[*] *Dynamica s.r.l. 26100 Cremona, Italy (e-mail:*
andrea.bartolini@dynamica-it.com).
[**] *Politecnico di Milano, Dipartimento di Elettronica, Informazione e*
Bioingegneria, 20133 Milano, Italy (e-mail:
francesco.casella,alberto.leva@polimi.it)

1. INTRODUCTION

The object-oriented modelling paradigm (see, e.g., Mattsson et al. (1998), Tiller (2001)) allows to build system models by connecting component or sub-system models in an arbitrary fashion through connectors, as long as the connection makes sense from a physical point of view, without worrying about the internal structure or implementation of the components.

Object-oriented languages and modelling tools have been used for a long time for the modelling of hydraulic systems, where pressurized oil is used to power mechanical equipment, see, e.g., the HyLib library (Beater, 2000), which later became the commercial HydraulicsLibrary, or other modelling libraries described by Harman (2006) and Paredis (2008). Several papers describe applications of these libraries to specific modelling needs, see, e.g., Harman (2006), or Chandrasekar and Tummescheit (2014).

In all these models, hydraulic dynamics is strongly coupled with mechanical dynamics. In principle, based on the time scale or frequency range of phenomena one is interested into for a specific simulation study, it would be possible to leave out the compressibility effects of the fluid and/or the inertia of some mechanical elements in some component models. In their recent experience, the authors have found out that allowing complete freedom in taking into account these phenomena can lead to severe numerical issues when system models are built out of them. To the author's knowledge, this aspect has never been explicitly discussed in the literature.

The goal of this paper is to explain these issues by means of a simple paradigmatic system model, and to provide guidelines for library developers to allow the arbitrary connection of component models, according to the object-oriented modelling principles.

2. THE PARADIGMATIC SYSTEM MODEL

The paradigmatic system is shown in Fig. 1. A rigid cylindrical chamber of section A containing a compressible fluid (usually oil) at pressure p, with nominal density ρ_0
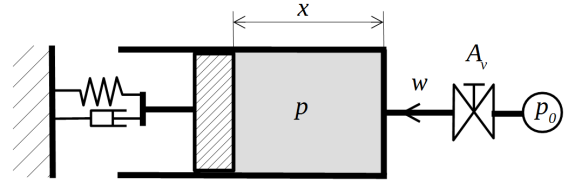

Fig. 1. Sketch of the paradigmatic system

at pressure p_0 and compressibility k_p, is delimited by a piston of mass m on the left side and by a valve with mass flow rate w and flow coefficient A_v on the right side. The piston is connected to a mechanical load with elastic coefficient k_x and friction coefficient k_v, which applies a force F on the piston that is the result of the elastic and friction effects plus an extra prescribed force F_0. The valve outlet is connected to a reservoir with fixed pressure p_0.

The system model is declaratively defined by the following set of differential-algebraic equations (DAEs).

$$A[\rho_0 + k_p(p - p_0)]\dot{x} + Axk_p\dot{p} - w = 0 \qquad (1)$$
$$\dot{x} - v = 0 \qquad (2)$$
$$m\dot{v} - Ap - F = 0 \qquad (3)$$
$$F - F_0 + k_x x + k_v v = 0 \qquad (4)$$
$$w - A_v\sqrt{\rho_0|p_0 - p|}\,sign(p_0 - p) = 0 \qquad (5)$$

where Eq. (1) is the mass balance in the chamber, Eqs. (2)-(3) describe the piston motion, Eq. (4) describes the mechanical load, and Eq. (5) the valve flow. F_0 and A_v are known system inputs, functions of time.

3. STRUCTURAL ANALYSIS

The system (1)-(5) is a DAE in the form $F(y, \dot{y}, v, t) = 0$; this DAE has index one if the Jacobian matrix $\frac{\partial F}{\partial z}$ (where z collects all terms in $\dot{y}$ and v) has a non-zero determinant, which means that the system can be locally solved for z given y and t.

If both fluid compressibility and mechanical inertia are taken into account, then $m > 0$ and $k_p > 0$. In this case, it turns out that

$$\left|\frac{\partial F}{\partial z}\right| = k_p mAx. \qquad (6)$$

As the cross section A is obviously positive, the system has index 1 provided that $x > 0$, i.e., the chamber is not completely empty of oil, which is a reasonable assumption for the validity of such a model.

When the load is much less stiff than the fluid, one may want to neglect the fluid compressibility, i.e., set $k_p = 0$, to get rid of one fast state in the model. In this case

$$\left|\frac{\partial F}{\partial z}\right| = \frac{A_v m}{2\sqrt{p_0 - p}} sign(p_0 - p) \qquad (7)$$

The determinant is undefined when $p = p_0$, but this is due to the slightly naive formulation of the valve equation (5), whose derivative with respect to p approaches infinity as $p \to p_0$. This problem can be solved by using a more appropriate formulation of that law that does not become singular at that point, e.g., by substituting the function $\sqrt{h}sign(h)$ with the function

$$\frac{h}{\sqrt[4]{h^2 + \epsilon^2}} \qquad (8)$$

which is close to the original function when $|h| \gg \epsilon$ but has a finite derivative in the neighbourhood of zero (Casella, 1999). Then

$$\left|\frac{\partial F}{\partial z}\right| = \frac{A_v m}{\sqrt{\epsilon}} \qquad (9)$$

In this case the index of the system is normally 1, but becomes 2 when the valve is closed ($A_v = 0$), which is a completely normal condition in hydraulic system models.

Alternatively, when the piston has a small mass (e.g., it is a membrane), one may want to neglect it, setting $m = 0$, but still accountinf for the fluid compressibility $k_p > 0$. In this case

$$\left|\frac{\partial F}{\partial z}\right| = k_p k_v xA \qquad (10)$$

Now, by assumption $k_p > 0$, $A > 0$, and $x > 0$ if the chamber is not empty. In principle, the model is index 1 if the viscous friction term k_v is non-zero. This is in general a critical assumption: for example, if one is modelling an accumulator with a membrane, it is very hard to identify a viscous friction term, as the reaction force of the membrane is inherently only depending on its deformation. If more sophisticated friction models were used, the condition equivalent to $k_v \neq 0$ would be that is is always possible to invert the force-velocity relationship, computing the force given the velocity, which for example not possible when modelling stiction.

In the extreme case, one may want to avoid the stiff dynamics entirely and neglect both fluid compressibility and inertia, thus setting $k_p = 0$ and $m = 0$. In this case

$$\left|\frac{\partial F}{\partial z}\right| = \rho_0 A^2 + \frac{A_v k_v}{\sqrt{\epsilon}} \qquad (11)$$

Since $A_v \geq 0$, and assuming $k_v \geq 0$, that is, if a viscous friction term is present, it is dissipative, then the system is always index 1. If more sophisticated friction models including stiction are used, it is impossible to say based on this analysis that the system will always be index 1, as they may entail $k_v < 0$ during transients. Further

analysis would be necessary, taking into account such friction models explicitly, but this goes beyond the scope of this paper.

4. DISCUSSION AND CONCLUSION

The analysis carried out in the previous section shows that in the two intermediate cases, and possibly in the last one, the system can change the index during simulation, which will cause a run-time error in existing equation-based, object-oriented modelling tools (EOOLTs), that cannot handle this case. This results gives a concise and clear explanation of the numerical problems that were experienced with real-life models taking the same modelling assumptions.

The results of the analysis suggest that the best option to guarantee the safe and robust simulation of equation-based, object-oriented hydraulic systems coupling fluid dynamics and mechanical dynamics is to always take into account both fluid compressibility and mechanical inertia of pistons and membranes, even though this may end up in modelling very stiff dynamics. Neglecting both fluid compressibility and mechanical inertia can be an option to reduce the number of fast modes in the system's dynamics, but only if simple enough (e.g. linear) models are used to represent friction phenomena. In the author's opinion, these findings are particularly useful for developers of reusable component libraries for hydraulic power and actuation systems.

REFERENCES

Beater, P. (2000). Modelling and digital simulation of hydraulic systems in design and engineering education using Modelica and HyLib. In *Proceedings of the Modelica Workshop 2000*, 1–8. Modelica Association, Lund, Sweden.

Casella, F. (1999). *Modelling, Simulation, and Control of a Geothermal Power Plant.* Ph.D. thesis, Politecnico di Milano.

Chandrasekar, S. and Tummescheit, H. (2014). Physical design of hydraulic valves in Modelica. In *Proceedings of the 10th International Modelica Conference*, 627–636. Lund, Sweden.

Harman, P. (2006). Modelling automotive hydraulic systems using the Modelica ActuationHydraulics library. In *Proceedings of the 5th International Modelica Conference*, 399–403. Vienna, Austria.

Mattsson, S.E., Elmqvist, H., and Otter, M. (1998). Physical system modeling with Modelica. *Control Engineering Practice*, 6(4), 501–510.

Paredis, C. (2008). An open-source Modelica library of fluid power models. In *Bath/ASME Symposium on Fluid Power and Motion Control*, 1–14. Bath, UK.

Tiller, M. (2001). *Introduction to physical modelling with Modelica.* Kluwer.