

Numerical Simulation of Dynamic Systems XVII

Prof. Dr. François E. Cellier
Department of Computer Science
ETH Zurich

April 23, 2013

The Solvability Issue

Let us look at the DAE:

$$x - \dot{x}^2 = 0$$

Converting to ODE form, we obtain:

$$\dot{x} = \pm\sqrt{x}$$

- ▶ We notice that the ODE has only a real-valued solution as long as the initial value of x is positive. This constraint exists implicitly also in the DAE formulation, but it is not directly visible.
- ▶ Yet, the problem is worse, because we don't know which root to choose. If we choose the positive root, \dot{x} will also be positive, and x will keep growing. However, if we choose the negative root, \dot{x} is negative, and x will decrease.
- ▶ Even worse, it could be that we should choose the positive root during some period of time, and the negative root during another. Thus, at any moment in time, we obtain a potential bifurcation in the solution depending on whether we choose the positive or the negative root.

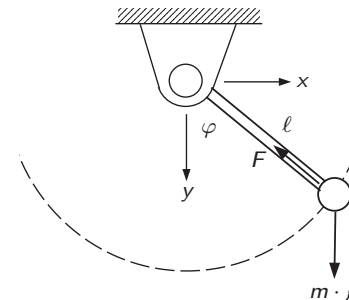
The Solvability Issue II

Does the solvability issue cause a real dilemma?

- ▶ **Physics does not provide us with unsolvable riddles.**
- ▶ Saying that a DAE is unsolvable is equivalent to saying that the phenomenon described by it is “defying causality” in the sense that the outcome of an experiment is non-deterministic, which in turn is almost equivalent to saying that the phenomenon is non-physical.
- ▶ Thus, if a DAE model contains solvability issues, this simply means that the DAE does not capture the physical phenomenon that it is supposed to describe in its full complexity. Some information is missing.
- ▶ Unfortunately, solvability issues are encountered frequently in DAE models derived from object-oriented descriptions of physical systems, and consequently, we need to deal with the consequences.

The Solvability Issue III

Let us look at a simple pendulum:



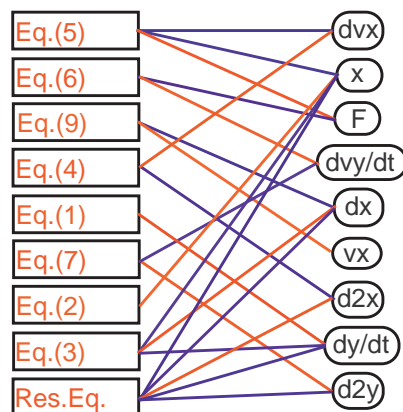
describable by the following set of DAEs:

$$\begin{aligned} m \cdot \frac{dv_x}{dt} &= -\frac{F \cdot x}{\ell} \\ m \cdot \frac{dv_y}{dt} &= m \cdot g - \frac{F \cdot y}{\ell} \\ \frac{dx}{dt} &= v_x \\ \frac{dy}{dt} &= v_y \\ x^2 + y^2 &= \ell^2 \end{aligned}$$

- ▶ Since x , y , v_x , and v_y are known state variables, the last equation in the set is a **constraint equation**.

The Solvability Issue VIII

Causalizing the remaining equations:



$$\begin{aligned}\frac{dy}{dt} &= v_y \\ x &= \pm \sqrt{\ell^2 - y^2} \\ dx &= -\frac{y}{x} \cdot \frac{dy}{dt} \\ dv_x &= d^2x \\ F &= -m \cdot \ell \cdot \frac{dv_x}{x} \\ \frac{dv_y}{dt} &= g - \frac{F \cdot y}{m \cdot \ell} \\ d^2y &= \frac{dv_y}{dt} \\ d^2x &= -\frac{dx^2 + \left(\frac{dy}{dt}\right)^2 + y \cdot d^2y}{x} \\ v_x &= dx\end{aligned}$$

The Solvability Issue IX

- ▶ The solution is formally correct. Our remaining state variables are y and v_y , and the resulting equations are thus perfectly causal except for the algebraic loop in the single tearing variable d^2x that needs to be solved by Newton iteration.
- ▶ Yet, we are encountering new problems.
- ▶ First, the simulation will blow up with a division by zero, as soon as $x = 0$.
- ▶ Second, we seem to have a *solvability issue*, as we don't know which of the two roots to choose.

The Solvability Issue X

- ▶ Physics doesn't have a "solvability issue" with the pendulum. The problem is purely mathematical. Somehow, our model does not contain full information.
- ▶ Full information had not even been available in the original DAE model, but the conversion to ODE form using the Pantelides and Tarjan algorithms has made the problem worse.
- ▶ In the DAE formulation, x had been a state variable, and consequently, the DAE model "knew" that the pendulum cannot jump. From the ODE model, that knowledge is no longer evident. The variable x could change its sign at any point in time, making the pendulum jump instantaneously from one side to the other.
- ▶ From our physical understanding, we know that we must choose the positive root for $x > 0$ and the negative root for $x < 0$. We also know that the pendulum will swing through $x = 0$, i.e., as we pass through zero, we need to switch to the other root. Yet, our mathematical description doesn't contain that information.

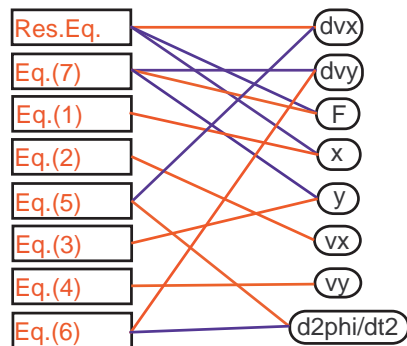
The Solvability Issue XI

- ▶ One way to get around the solvability issue in the case of the pendulum system is to *select another set of state variables*.
- ▶ A far better choice of state variables would have been the angle φ and the angular velocity $\dot{\varphi}$.
- ▶ Unfortunately, these variables don't appear in our previous model at all.
- ▶ For this reason, it may be best to reformulate the original DAE model.

$$\begin{aligned}m \cdot \frac{dv_x}{dt} &= -\frac{F \cdot x}{\ell} \\ m \cdot \frac{dv_y}{dt} &= m \cdot g - \frac{F \cdot y}{\ell} \\ \frac{dx}{dt} &= v_x \\ \frac{dy}{dt} &= v_y \\ x &= \ell \cdot \sin(\varphi) \\ y &= \ell \cdot \cos(\varphi)\end{aligned}$$

The Solvability Issue XVI

We choose a tearing variable and a residual equation and finish causalization:



$$\begin{aligned}
 x &= \ell \cdot \sin(\varphi) \\
 v_x &= \ell \cdot \cos(\varphi) \cdot \frac{d\varphi}{dt} \\
 y &= \ell \cdot \cos(\varphi) \\
 v_y &= -\ell \cdot \sin(\varphi) \cdot \frac{d\varphi}{dt} \\
 \frac{d^2\varphi}{dt^2} &= \frac{dv_x}{\ell \cdot \cos(\varphi)} + \frac{\sin(\varphi)}{\cos(\varphi)} \cdot \left(\frac{d\varphi}{dt}\right)^2 \\
 dv_y &= -\ell \cdot \sin(\varphi) \cdot \frac{d^2\varphi}{dt^2} - \ell \cdot \cos(\varphi) \cdot \left(\frac{d\varphi}{dt}\right)^2 \\
 F &= \frac{m \cdot g \cdot \ell}{y} - \frac{m \cdot \ell \cdot dv_y}{y} \\
 dv_x &= -\frac{F \cdot x}{m \cdot \ell}
 \end{aligned}$$

The Solvability Issue XVII

- ▶ Our remaining state variables are φ and $\frac{d\varphi}{dt}$, as desired, and the resulting equations are causal except for one algebraic loop in the single tearing variable dv_x that needs to be solved by Newton iteration.
- ▶ The former *solvability issue* is gone. We no longer have to choose between a positive and a negative root.
- ▶ Unfortunately, we are still facing a *singularity issue*. This is not a *structural singularity*, but rather a *dynamic singularity*.
- ▶ The simulation will work fine, as long as we don't let the pendulum swing beyond the horizontal position.
- ▶ Unfortunately, as $y = 0$, the simulation will once again blow up with a division by zero.

The Solvability Issue XVIII

- ▶ Many non-linear mechanical multi-body systems exhibit dynamic singularity issues when their dynamics are described in an object-oriented fashion by a set of DAEs.
- ▶ **Dymola** recognizes potential singularity issues during compilation. In such cases, Dymola will keep additional state variables in the set of equations and perform a *dynamic state selection*, i.e., Dymola will switch dynamically to another set of state variables on the fly as the current set approaches one of its singular points.

Conclusions

- ▶ In the previous three presentations on converting sets of DAEs to equivalent sets of ODEs, we concentrated for simplicity on *linear electric circuits* as examples, as these models are easily understandable.
- ▶ However, some issues don't show up in linear systems. In this final presentation on the symbolic preprocessing of DAE systems, we focused on precisely those remaining issues that can occur only in *non-linear systems*: the solvability issue and the dynamic singularity issue.
- ▶ **Dymola** recognizes *solvability issues* during compilation. To this end, Dymola avoids whenever possible to make use of state variables whose derivatives are to be solved from an equation, in which they appear in a non-linear form. Similarly, Dymola avoids to select tearing variables that appear in their residual equations in a non-linear form.

Conclusions II

- **Dymola** also recognizes *dynamic singularity issues* during compilation. To this end, Dymola avoids whenever possible to make use of state variables whose derivatives are multiplied by other variables in the equations from which they need to be solved, as these other variables would invariably turn up in the denominator after the symbolic manipulation. Similarly, Dymola avoids to select tearing variables that are multiplied by other variables in their residual equations. When this cannot be avoided, Dymola keeps additional state variables and/or additional tearing variables in the set of iteration variables.