Partial Evaluation of AD for DAE Solvers*

O. Eriksson, V. Palmkvist, and D. Broman

*published in GPCE 2023

Differential Algebraic Equations (DAEs)

Differential equations

$$\frac{d^2x(t)}{dt^2} = \lambda(t)x(t)$$

$$\frac{d^2y(t)}{dt^2} = \lambda(t)y(t) - G$$

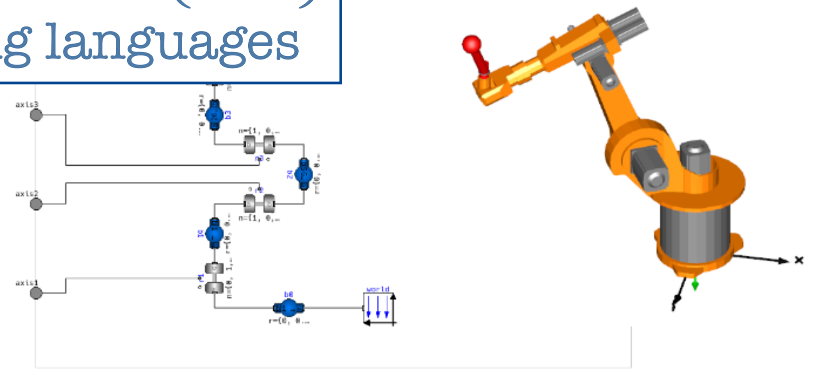
$$L^2 = x(t)^2 + y(t)^2$$

Algebraic constraint

Model from first principles

MapleSim
Simscape
Dymola
OpenModelica
Modelica Language

Foundation of Equation-Based Object-Oriented (EOO) modelling languages



*From the Modelica standard library

EOO Language Toolchain Challenges

High-performance simulation

↓

Numerical integration

Index-reduction

$$\frac{d^{c_1}}{dt^{c_1}} f_1\left(\dot{\boldsymbol{x}}(t), \boldsymbol{x}(t)\right) = 0$$

$$\frac{d^{c_2}}{dt^{c_2}} f_2\left(\dot{\boldsymbol{x}}(t), \boldsymbol{x}(t)\right) = 0$$

$$\vdots$$

$$\frac{d^{c_n}}{dt^{c_n}} f_n\left(\dot{\boldsymbol{x}}(t), \boldsymbol{x}(t)\right) = 0$$

f_i after index reduction and order reduction

$$\boldsymbol{J}_{ij} = \frac{\partial f_i}{\partial \dot{\boldsymbol{x}}_j} + \frac{\partial f_i}{\partial \boldsymbol{x}_j}$$

Jacobian

$$\dot{\boldsymbol{x}}(t) = \frac{d\boldsymbol{x}(t)}{dt}$$

Index-reduction and Jacobian evaluation in EOOs

$$\mathscr{D}_x[\sin(x)] = \cos(x)$$

$$\mathscr{D}_{AD}[\lambda x \,.\, \sin(x)] = \lambda y \,.\, ((\lambda(x, \delta x) \,.\, (\sin(x), \delta x \cdot \cos(x))\ (y, 1)).1$$

Forward-mode because our problem is square

Symbolic Differentiation (SD)

+ Efficient code

- n^2 symbolic $\frac{\partial f_i}{\partial \dot{\boldsymbol{x}}_j} + \frac{\partial f_i}{\partial \boldsymbol{x}_j}$

- Not easy for general programs

Automatic Differentiation (AD)

+ Natural for general programs

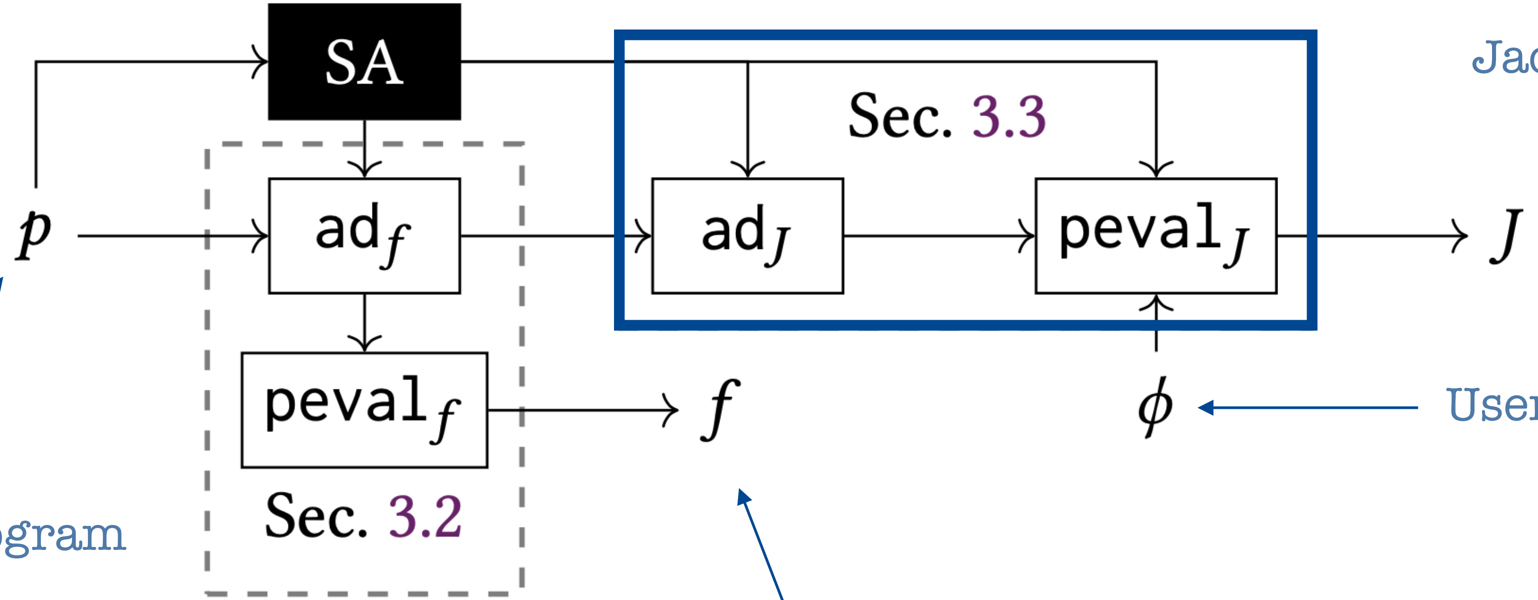
+ $\boldsymbol{J}_j$ size prop to/complexity of f

- Indirection/tupling/projection

Our Approach: Combine AD and SD for DAEs via PE

Forward-mode PEAD for DAEs

$$\texttt{peval}[\mathscr{D}_{AD}[\lambda x\,.\,\sin(x)]] = \lambda y\,.\,\cos(y)$$



Jacobian callback

User parameter

High-index DAE program

Low-index residual callback

$$p \in \texttt{Prog} ::= \textbf{equations}\ (e_0, \ldots, e_n) \mid \textbf{let}\ x = e\ \textbf{in}\ p \mid \textbf{letrec}\ x = e\ \textbf{in}\ p$$

Part of DAE DSL in Miking Framework

Illustration of Parametrised Jacobian Specialisation

$n = 6$

$$\begin{pmatrix} \bullet & 0 & \bullet & 0 & 0 & 0 \\ 0 & \bullet & \bullet & 0 & 0 & 0 \\ \bullet & \bullet & 0 & 0 & \bullet & \bullet \\ 0 & 0 & 0 & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet \end{pmatrix}$$

$\phi n = 0$

Specialised columns

Pure AD columns

0 structurally zero $\bullet$ structurally non-zero

Illustration of Parametrised Jacobian Specialisation

$n = 6$

$$\begin{pmatrix} \bullet & 0 & \bullet & 0 & 0 & 0 \\ 0 & \bullet & \bullet & 0 & 0 & 0 \\ \bullet & \bullet & 0 & 0 & \bullet & \bullet \\ 0 & 0 & 0 & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet \end{pmatrix}$$

$\phi n = 1$

Specialised columns

Pure AD columns

0 structurally zero $\bullet$ structurally non-zero

Illustration of Parametrised Jacobian Specialisation

$n = 6$

$$\left(\begin{array}{cccc|cc} \bullet & 0 & \bullet & 0 & 0 & 0 \\ 0 & \bullet & \bullet & 0 & 0 & 0 \\ \bullet & \bullet & 0 & 0 & \bullet & \bullet \\ 0 & 0 & 0 & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet \end{array}\right)$$

$\phi n = 2$

Specialised columns (green box)

Pure AD columns (red box)

0 structurally zero $\bullet$ structurally non-zero

Illustration of Parametrised Jacobian Specialisation

$n = 6$

$$\begin{pmatrix} \bullet & 0 & \bullet & 0 & 0 & 0 \\ 0 & \bullet & \bullet & 0 & 0 & 0 \\ \bullet & \bullet & 0 & 0 & \bullet & \bullet \\ 0 & 0 & 0 & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet \end{pmatrix}$$

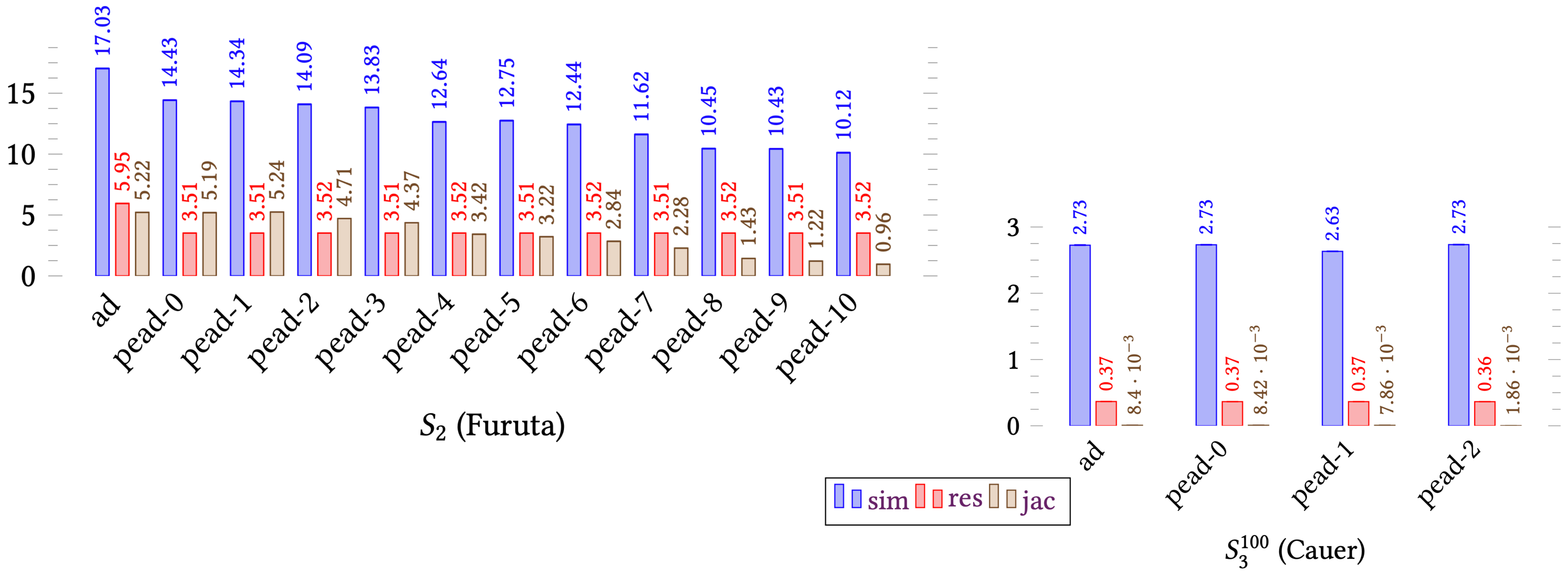
$\phi n = 3$

Specialised columns

Pure AD columns

0 structurally zero $\bullet$ structurally non-zero

Experimental Results (median execution time [s], 25 runs)



Conclusion

We design a technique for PE of AD in a DAE context

It applies to both index-reduction and Jacobian evaluation

The evaluation shows that improves performance of non-linear models compared to a pure AD approach