POLITECNICO DI MILANO

Multibody System Dynamics: MBDyn Hydraulics Modeling

Pierangelo Masarati <masarati@aero.polimi.it>
Politecnico di Milano
Dipartimento di Ingegneria Aerospaziale

Outline

MBDyn

- Introduction
- Modeling Scales
- Multibody/Multiphysics Dynamics:
    - MBDyn Software
    - Modeling Approach
- Hydraulic Library Overview
- Examples

Introduction

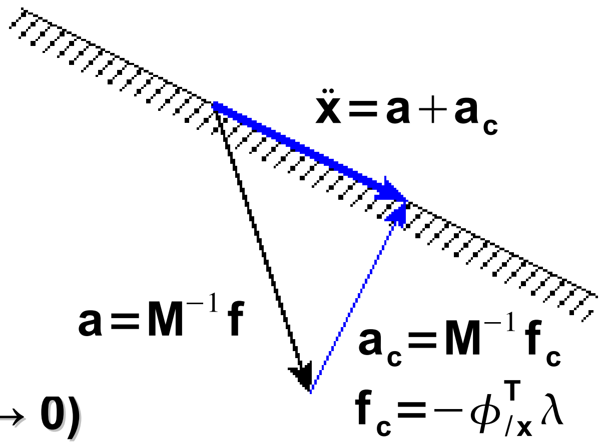
MBDyn

- **Multibody Dynamics: unconstrained mechanical systems...**

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}}=\mathbf{f}(\mathbf{x},\dot{\mathbf{x}},\mathbf{t})$$

- **… plus kinematic constraints: constrained mechanical systems**

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}}+\phi_{/\mathbf{x}}^{\mathbf{T}}\lambda=\mathbf{f}(\mathbf{x},\dot{\mathbf{x}},\mathbf{t})$$
$$\phi(\mathbf{x},\mathbf{t})=0$$



- **System of differential-algebraic equations (DAE)**

- **Infinitely "fast" dynamics (time scale → 0)**

- **Requires unconditionally stable integration, algorithmic dissipation → implicit (nearly) L-stable schemes**

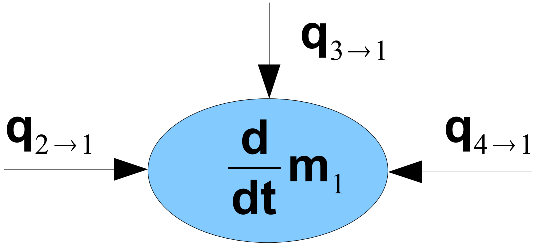
Introduction

MBDyn

- **Hydraulic system dynamics:**
  - **neglect local dynamics, spatial resolution: 0D & 1D**
  - **circuit theory: node pressures, branch flows**

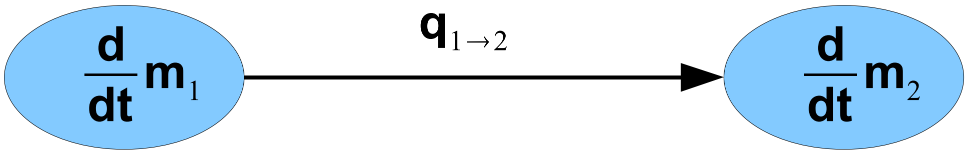
- **Flow balance at nodes**

$$\sum q_b = \frac{d}{dt} m_n$$



- **Constitutive properties of branches (e.g. pressure loss)**

$$\psi(q_b, \dot{q}_b, p_{n1}, \dot{p}_{n1}, p_{n2}, \dot{p}_{n2}, t) = 0$$



Introduction

MBDyn

- **Constitutive properties of fluid: linearization about reference condition**

$$\frac{d}{dt}m = \frac{d}{dt}\left(\rho V\right) = \dot{\rho} V + \rho \dot{V}$$

volume change (e.g. actuator) → $\rho \dot{V}$

$$\dot{\rho} = \frac{\partial \rho}{\partial p}\dot{p} + \frac{\partial \rho}{\partial T}\dot{T} = \rho_0\left(\frac{1}{\beta}\dot{p} - \alpha \dot{T}\right)$$

compressibility ($\frac{1}{\beta}\dot{p}$) **can be neglected if:**
- **m small**
- **bulk modulus large**
- **pressure rate small**

thermal expansion ($\alpha \dot{T}$) **can be neglected if:**
- **m small**
- **coefficient small**
- **temperature rate small**

$$\frac{d}{dt}m = \frac{m}{\beta}\dot{p} - \xcancel{m\alpha\dot{T}} + \rho \dot{V}$$

assume temperature rate small compared to time scale of hydro-mechanical processes

Modeling Scales

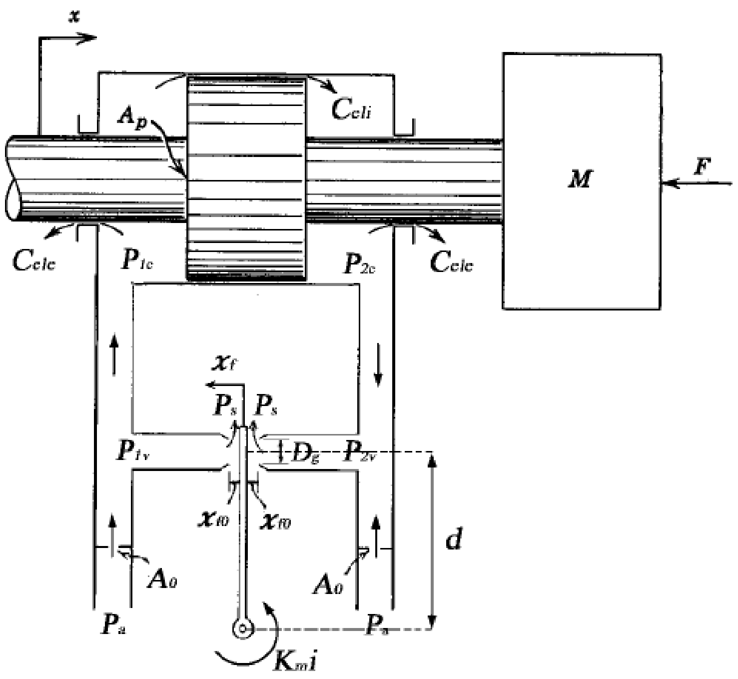
MBDyn

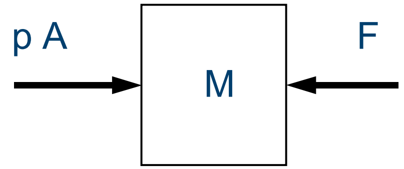
- Multiphysics problem: interaction between different domains (e.g. mechanical and hydraulic)
- Interaction described in terms of:
  - Frequency, bandwidth (how rapid phenomena are)
  - Power (how much work is transfered between domains)
- Determine how interaction can be simplified:
  - Truncation
  - Steady approximation
  - Quasi-steady approximation
  - Full coupling

Modeling Scales

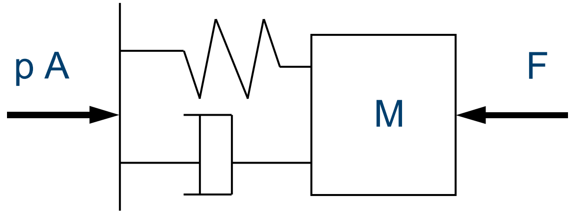
MBDyn

- **Example: actuator**





truncated model: commanded force independed from dynamics



(quasi-)static model: commanded force depends on dynamics approx. as mass-spring-damper

Approximations

MBDyn

- **Coupled dynamic problem: state-space representation**

$$\dot{x} = A\,x + B\,u$$
$$y = C\,x + D\,u$$

- **States partitioned based on frequency separation ("slow" vs. "fast"):**

$$\begin{Bmatrix} \dot{x}_s \\ \dot{x}_f \end{Bmatrix} = \begin{bmatrix} A_{ss} & A_{sf} \\ A_{fs} & A_{ff} \end{bmatrix} \begin{Bmatrix} x_s \\ x_f \end{Bmatrix} + \begin{bmatrix} B_s \\ B_f \end{bmatrix} u$$
$$y = \begin{bmatrix} C_s & C_f \end{bmatrix} \begin{Bmatrix} x_s \\ x_f \end{Bmatrix} + D\,u$$

- **Approximations consist in reducing the system to the "slow" dynamics while preserving information about the "fast" dynamics**

Approximations

MBDyn

- **Truncation: only consider “slow” states**

$$\begin{Bmatrix} \dot{\mathbf{x}}_s \\ 0 \end{Bmatrix} = \begin{bmatrix} \mathbf{A}_{ss} & \mathbf{A}_{sf} \\ \mathbf{A}_{fs} & \mathbf{A}_{ff} \end{bmatrix} \begin{Bmatrix} \mathbf{x}_s \\ 0 \end{Bmatrix} + \begin{bmatrix} \mathbf{B}_s \\ \mathbf{B}_f \end{bmatrix} \mathbf{u}$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{C}_s & \mathbf{C}_f \end{bmatrix} \begin{Bmatrix} \mathbf{x}_s \\ 0 \end{Bmatrix} + \mathbf{D}\mathbf{u}$$

- **Reduced system:**

$$\dot{\mathbf{x}}_s = \mathbf{A}_{ss}\mathbf{x}_s + \mathbf{B}_s\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}_s\mathbf{x}_s + \mathbf{D}\mathbf{u}$$

Approximations

MBDyn

- **Steady approximation: only statically consider "fast" states**

$$\begin{Bmatrix} \dot{\mathbf{x}}_s \\ 0 \end{Bmatrix} = \begin{bmatrix} \mathbf{A}_{ss} & \mathbf{A}_{sf} \\ \mathbf{A}_{fs} & \mathbf{A}_{ff} \end{bmatrix} \begin{Bmatrix} \mathbf{x}_s \\ \mathbf{x}_f \end{Bmatrix} + \begin{bmatrix} \mathbf{B}_s \\ \mathbf{B}_f \end{bmatrix} \mathbf{u}$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{C}_s & \mathbf{C}_f \end{bmatrix} \begin{Bmatrix} \mathbf{x}_s \\ \mathbf{x}_f \end{Bmatrix} + \mathbf{D}\mathbf{u}$$

- **Reduced system:**

$$\dot{\mathbf{x}}_s = \left(\mathbf{A}_{ss} - \mathbf{A}_{sf}\mathbf{A}_{ff}^{-1}\mathbf{A}_{fs}\right)\mathbf{x}_s + \left(\mathbf{B}_s - \mathbf{A}_{sf}\mathbf{A}_{ff}^{-1}\mathbf{B}_f\right)\mathbf{u}$$

$$\mathbf{x}_f = -\mathbf{A}_{ff}^{-1}\mathbf{A}_{fs}\mathbf{x}_s - \mathbf{A}_{ff}^{-1}\mathbf{B}_f\mathbf{u}$$

$$\mathbf{y} = \left(\mathbf{C}_s - \mathbf{C}_f\mathbf{A}_{ff}^{-1}\mathbf{A}_{fs}\right)\mathbf{x}_s + \left(\mathbf{D} - \mathbf{C}_f\mathbf{A}_{ff}^{-1}\mathbf{B}_f\right)\mathbf{u}$$

- **Note: the original system becomes differential-algebraic (DAE)**

Approximations

MBDyn

- **Quasi-steady approximation: use low-order dynamics**
- **In Laplace's domain:**

$$\mathbf{y}(\mathbf{s})=\left(\mathbf{C}(\mathbf{s}\mathbf{I}-\mathbf{A})^{-1}\mathbf{B}+\mathbf{D}\right)\mathbf{u}(\mathbf{s})=\mathbf{H}(\mathbf{s})\mathbf{u}(\mathbf{s})$$

$$\mathbf{y}(\mathbf{s})\simeq\left(\mathbf{H}(0)+\mathbf{s}\left(\frac{d\mathbf{H}}{d\mathbf{s}}\right)_{\mathbf{s}=0}+\frac{\mathbf{s}^2}{2}\left(\frac{d^2\mathbf{H}}{d\mathbf{s}^2}\right)_{\mathbf{s}=0}+\ldots+\frac{\mathbf{s}^\mathbf{n}}{\mathbf{n}!}\left(\frac{d^\mathbf{n}\mathbf{H}}{d\mathbf{s}^\mathbf{n}}\right)_{\mathbf{s}=0}\right)\mathbf{u}(\mathbf{s})$$

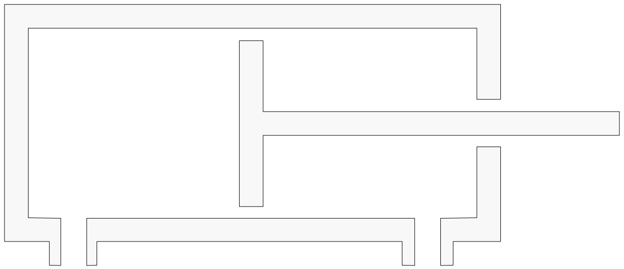
$$\mathbf{H}(0)=-\mathbf{C}\mathbf{A}^{-1}\mathbf{B}+\mathbf{D}$$
$$\mathbf{H}'(0)=-\mathbf{C}\mathbf{A}^{-2}\mathbf{B}$$
$$\mathbf{H}''(0)=-2\mathbf{C}\mathbf{A}^{-3}\mathbf{B}$$
$$\ldots$$
$$\mathbf{H}^{(\mathbf{n})}(0)=-(\mathbf{n}!)\mathbf{C}\mathbf{A}^{-(\mathbf{n}+1)}\mathbf{B}$$

- **Back to time domain:**

$$\mathbf{y}(\mathbf{t})=\mathbf{H}(0)\mathbf{u}(\mathbf{t})+\mathbf{H}'(0)\dot{\mathbf{u}}(\mathbf{t})+\frac{1}{2}\mathbf{H}''(0)\ddot{\mathbf{u}}(\mathbf{t})+\ldots$$

Approximations

MBDyn



Example: actuator; equations:

- **Flow balance in chamber 1**
- **Flow balance in chamber 2**
- **Equilibrium of piston**

$$V_1/\beta \dot{p}_1 = -A_p \dot{x} - C_{eli} A_{eli} (p_1 - p_2) - C_{ele} A_{ele} p_1 - C_{els} A_{els} (p_s - p_1)$$
$$V_2/\beta \dot{p}_2 = +A_p \dot{x} - C_{eli} A_{eli} (p_2 - p_1) - C_{ele} A_{ele} p_2 - C_{elr} A_{elr} (p_2 - p_r)$$
$$m \ddot{x} = A_p (p_1 - p_2) - r \dot{x} + F$$

- **State-space realization**

$$\begin{Bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \dot{x} \\ \dot{v} \end{Bmatrix} = \begin{bmatrix} -C_{e11}\beta/V_1 & C_{e12}\beta/V_1 & 0 & -A_p\beta/V_1 \\ C_{e12}\beta/V_2 & -C_{e22}\beta/V_2 & 0 & A_p\beta/V_2 \\ 0 & 0 & 0 & 1 \\ A_p/m & -A_p/m & 0 & -r/m \end{bmatrix} \begin{Bmatrix} p_1 \\ p_2 \\ x \\ v \end{Bmatrix} + \begin{Bmatrix} C_{e10}\beta/V_1 p_s \\ C_{e20}\beta/V_2 p_r \\ 0 \\ F/m \end{Bmatrix}$$

Approximations

MBDyn

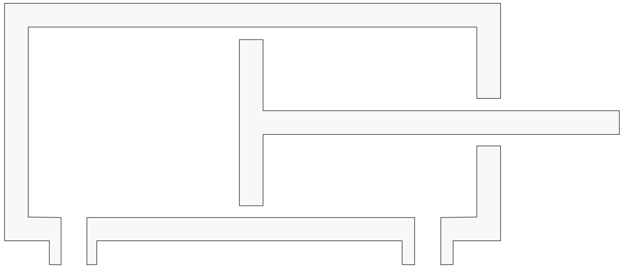
Truncation approximation:

- **Neglect hydraulics (velocities small, infinite power available)**

$$m\ddot{x}=A_p\left(p_s-p_r\right)-r\dot{x}+F$$

- **Directly control the force exerted by fluid**

$$\left\{\begin{matrix}\dot{x}\\ \dot{v}\end{matrix}\right\}=\begin{bmatrix}0 & 1\\ 0 & -r/m\end{bmatrix}\left\{\begin{matrix}x\\ v\end{matrix}\right\}+\left\{\begin{matrix}0\\ A_p/m\left(p_s-p_r\right)+F/m\end{matrix}\right\}$$



Static approximation:

- **Neglect compressibility (volume is small, bulk modulus is high)**

$$0\simeq-A_p\dot{x}-C_{eli}A_{eli}\left(p_1-p_2\right)-C_{ele}A_{ele}p_1-C_{els}A_{els}\left(p_s-p_1\right)$$
$$0\simeq+A_p\dot{x}-C_{eli}A_{eli}\left(p_2-p_1\right)-C_{ele}A_{ele}p_2-C_{elr}A_{elr}\left(p_2-p_r\right)$$
$$m\ddot{x}=A_p\left(p_1-p_2\right)-r\dot{x}+F$$

- **Pressure depends on velocity: → equivalent damping**

$$\left\{\begin{matrix}\dot{x}\\ \dot{v}\end{matrix}\right\}=\begin{bmatrix}0 & 1\\ 0 & -r/m-c_i/m\end{bmatrix}\left\{\begin{matrix}x\\ v\end{matrix}\right\}+\left\{\begin{matrix}0\\ A_p c_e/m\left(p_s-p_r\right)+F/m\end{matrix}\right\}$$

Software

MBDyn

- AMESim:
  - Monolithic software (can be interfaced as dynamic module to other solvers)
  - Models hydraulic networks in detail
- ADAMS:
  - Module for basic solver
  - Introduces modeling capabilities of hydraulic components
- Modelica (Dymola, MathModelica, OpenModelica?)
  - Modeling language, based on open library of elements
  - Broad library for general-purpose components
  - Many fields, including hydraulics and multibody
  - Needs specific (closed) solver
- MBDyn

Software

MBDyn

- MBDyn:
- Developed at Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano
- Monolithic multibody/multiphysics general-purpose software
- It's free: released under GPL (GNU General Public License)
- Includes integrated hydraulic library

Software

MBDyn

- Web site: http://www.aero.polimi.it/mbdyn/
- Distributed in source form
- Developed for Linux
- Needs Un*x-like build environment
- Binaries compiled for Windows XP available here:
  http://www.aero.polimi.it/masarati/Download/mbdyn/
- Windows executable has trouble with DOS-like line terminators;
  use http://www.aero.polimi.it/masarati/pe/es2/dos2unix.exe

Software

MBDyn

- Command-line software
- Prepare an input file using your favourite text editor
- Execute:

```
# mbdyn -f input_file -o output_file_prefix
```

- Output in files with specific extensions
- Load output files in math environment (octave, scilab, matlab, …) for plotting and further manipulation

Software

MBDyn

- Output can be reformatted for some post-processing tools
  - built-in: EasyAnim
  - ...
- Ongoing third-party project (“Blender & MBDyn”) about using Blender http://www.blender.org/ for pre/post-processing:

  http://www.baldwintechnology.com/

Software

MBDyn

- The model and the analysis are defined in an input file
- Use your preferred editor to prepare the input file
- The structure and the syntax of the statements are described here
  http://www.aero.polimi.it/mbdyn/download/
  (pick the manual for the version in use, or follow instructions)
- A set of tutorials is presented here
  http://www.aero.polimi.it/mbdyn/documentation/tutorials/

Modeling Approach

MBDyn

- **Nodes instantiate degrees of freedom and the corresponding balance equations**

- **Static structural nodes only instantiate equilibrium equations**

$$0 = \sum \mathbf{f}$$
$$0 = \sum \mathbf{m}$$

- **Dynamic structural nodes also instantiate momentum and momenta moment definitions**

$$\mathbf{M}\dot{\mathbf{x}} = \beta$$
$$\mathbf{J}\omega = \gamma$$
$$\dot{\beta} = \sum \mathbf{f}$$
$$\dot{\gamma} = \sum \mathbf{m}$$

- **Hydraulic nodes (pressure) instantiate flow balance equations**

Modeling Approach

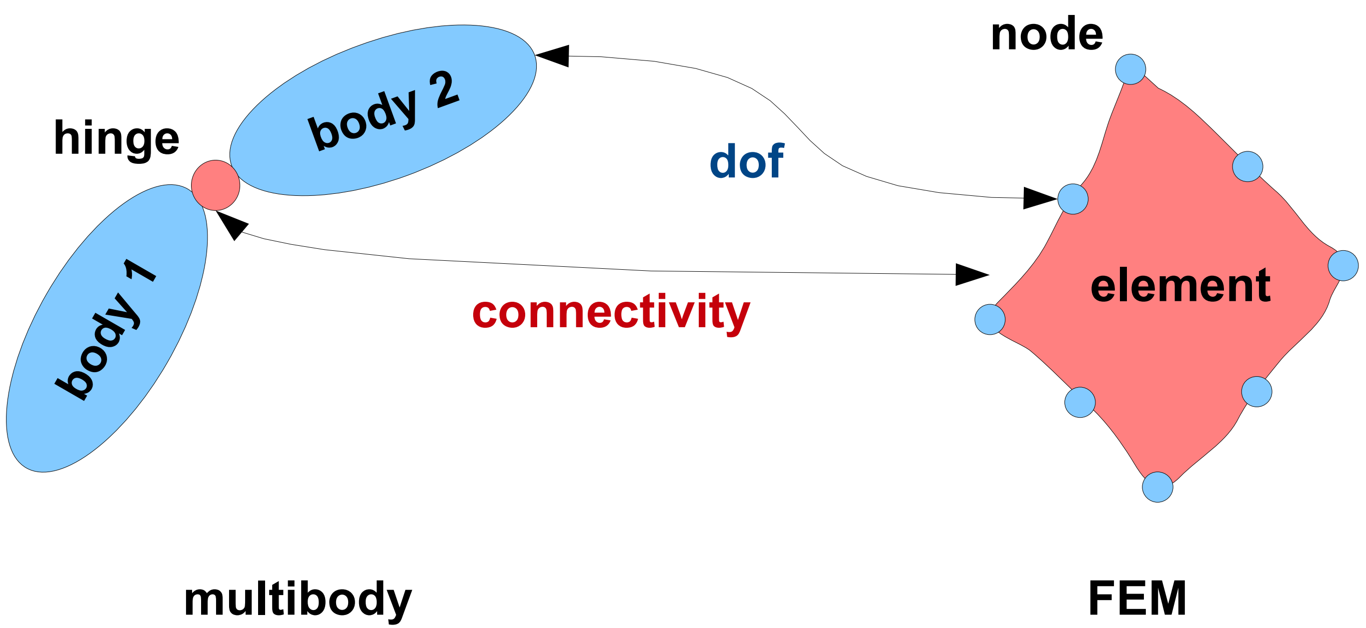
MBDyn

- Elements write contributions to nodes equations
- Elements represent “connectivity” and “constitutive properties”
- Elements can add further, “private” equations (e.g. algebraic constraints)
- Mechanical elements typically add forces and moments to equilibrium equations
- Mechanical constraints (“joints”) may add algebraic relationships between kinematic degrees of freedom
- Hydraulic elements typically add flow contributions to flow balance equations in nodes

Modeling Approach

MBDyn

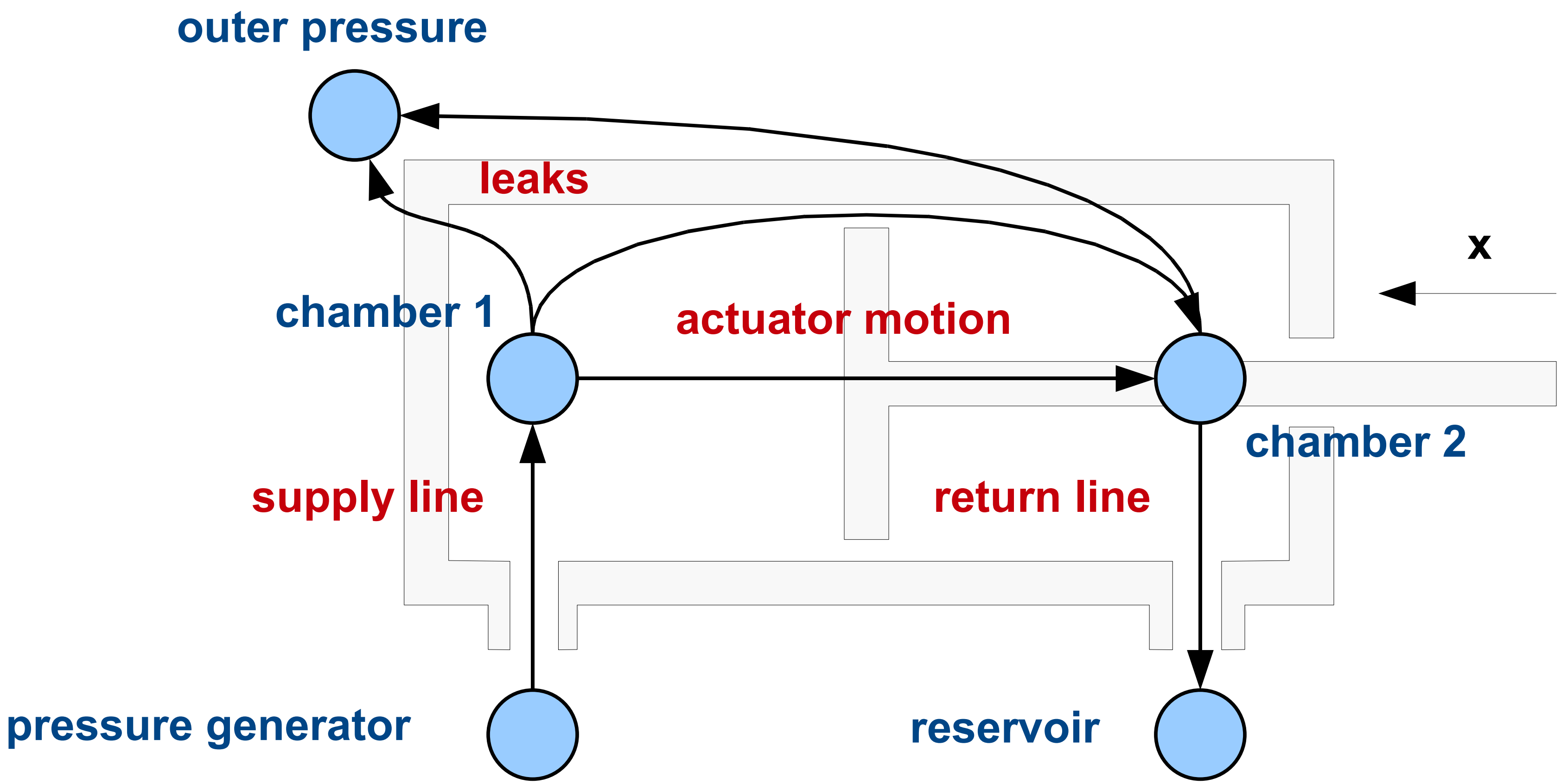
- **Multibody vs. FEM: nodes at bodies vs. nodes at frontier (node ↔ body; element ↔ hinge)**



Modeling Approach

MBDyn

- **Hydraulic modeling: network**



Hydraulic Element Library: Pressure Generator

MBDyn

- Not specifically implemented; use a “clamp” genel instead
- This element adds a scalar algebraic equation that enforces a specific value (possibly time-dependent) on a scalar node

$$\mathbf{p}=\mathbf{p}_0(\mathbf{t})$$

- This algebraic equation implies a Lagrange multiplier on the flow balance equation related to the pressure node

$$\sum \mathbf{q}+\lambda=0$$

- The multiplier corresponds to the flow required to grant the imposed pressure value

Hydraulic Element Library: Imposed Flow

MBDyn

- Not specifically implemented, use an “abstract” force instead
- This element adds a contribution, possibly time-dependent, to an arbitrary scalar equation

$$q = q(t)$$

- Note: a negative flow enters the circuit at the given node, a positive flow leaves the circuit at the given node

Hydraulic Element Library: Dynamic Pipeline

MBDyn

- Dynamic pipes formulated by a finite-volume approach
- Degenerate into static when pressure time derivatives neglected
- Differential mass balance:

$$\frac{\mathbf{D}}{\mathbf{Dt}}\mathbf{dm}=0 \rightarrow \mathbf{q}_{/\mathbf{x}}+\mathbf{A}\,\rho_{/\mathbf{t}}=0$$

- Differential momentum balance:

$$\frac{\mathbf{D}}{\mathbf{Dt}}\mathbf{dq}=\mathbf{df} \rightarrow \mathbf{q}_{/\mathbf{t}}+\left(\frac{\mathbf{q}^2}{\rho\,\mathbf{A}}+\mathbf{A}\,\mathbf{p}\right)_{/\mathbf{x}}=\mathbf{f}_{\mathbf{v}}$$

- Discretization:

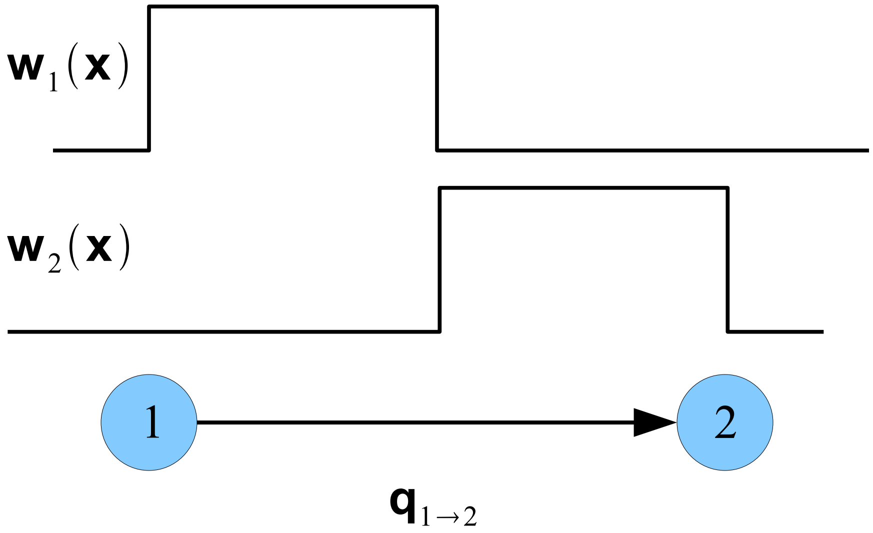
$$\mathbf{q}(\mathbf{x})=\begin{bmatrix}\frac{1-\xi}{2} & \frac{1+\xi}{2}\end{bmatrix}\begin{Bmatrix}\mathbf{q}_1\\ \mathbf{q}_2\end{Bmatrix}$$

$$\mathbf{p}(\mathbf{x})=\begin{bmatrix}\frac{1-\xi}{2} & \frac{1+\xi}{2}\end{bmatrix}\begin{Bmatrix}\mathbf{p}_1\\ \mathbf{p}_2\end{Bmatrix}$$

Hydraulic Element Library: Dynamic Pipeline

MBDyn

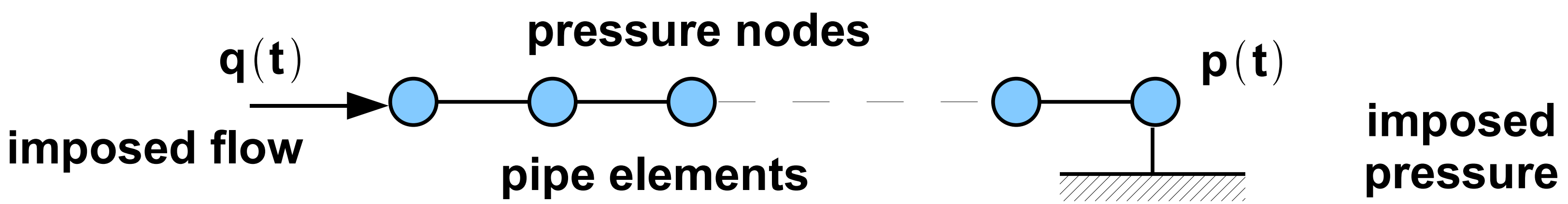
- **Piecewise-constant weight functions:**
- **Weight the mass and momentum balance equations**
- **Four constitutive equations in pressure and flow at nodes**



Example: Pressure Wave in Pipeline

MBDyn

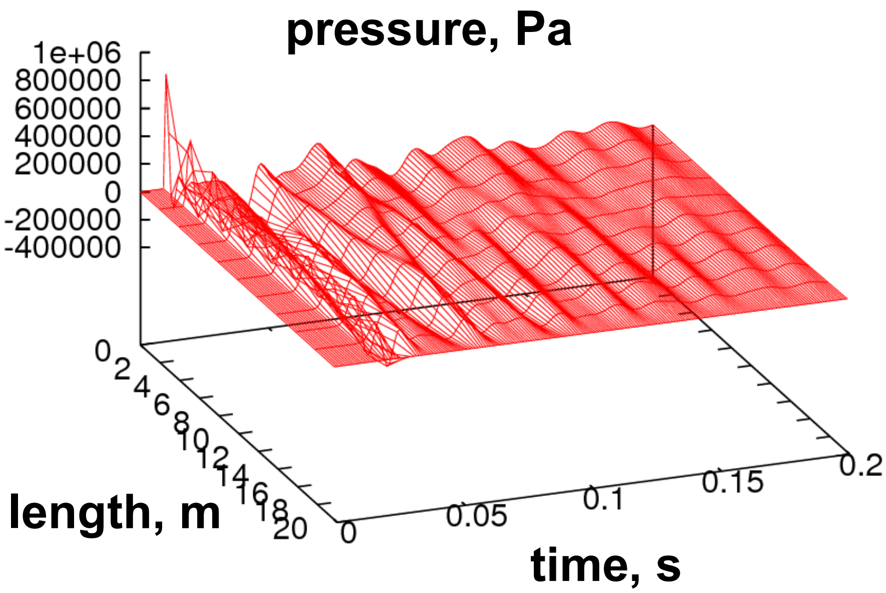
- From: R. Piché, A. Ellman, "A Fluid Transmission Line Model for Use with ODE Simulators", 8th Bath International Fluid Power Workshop, Sep. 20-22 1995, University of Bath, UK
  - Length: 19.74 m
  - Radius: 6.17e-3 m
  - Density: 870 kg/m^3
  - Viscosity: 8.e-5 m^2/s
  - Sound celerity: 1.4e3 m/s
  - Impulsive flow: 0.001 m^3/s
  - Duration: 0.1e-3 s
- Impulsive dynamics

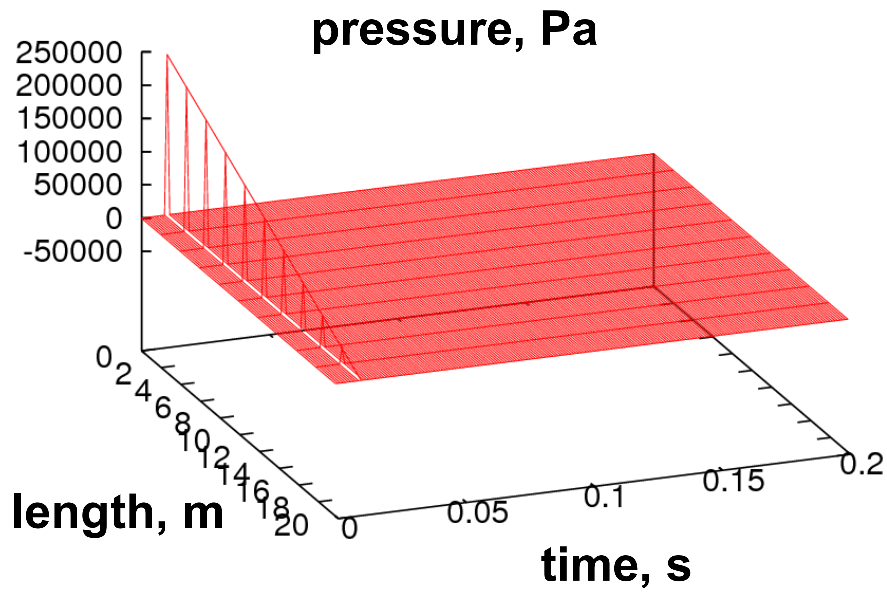


Example: Pressure Wave in Pipeline

MBDyn

- Dynamic pipe (10 “dynamic pipe” elements) compared to static pipe:

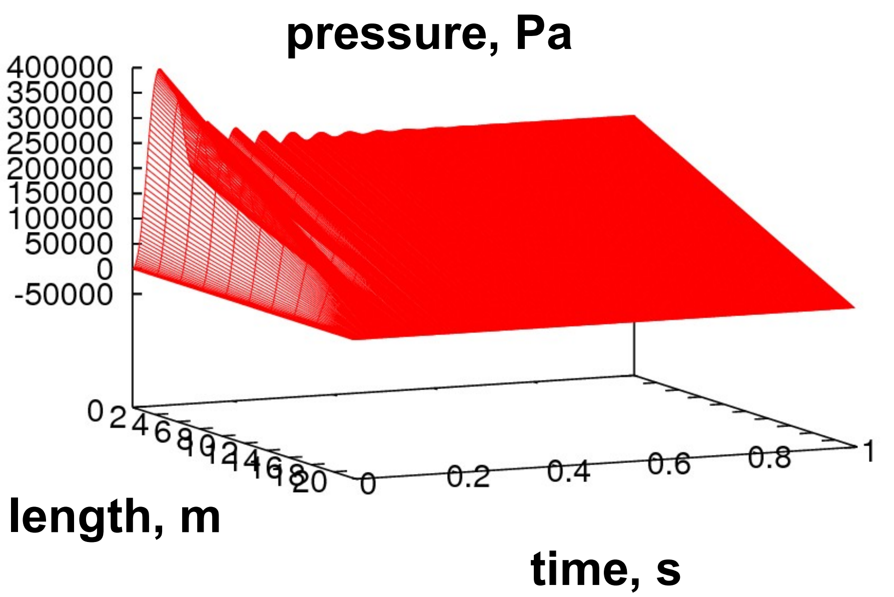


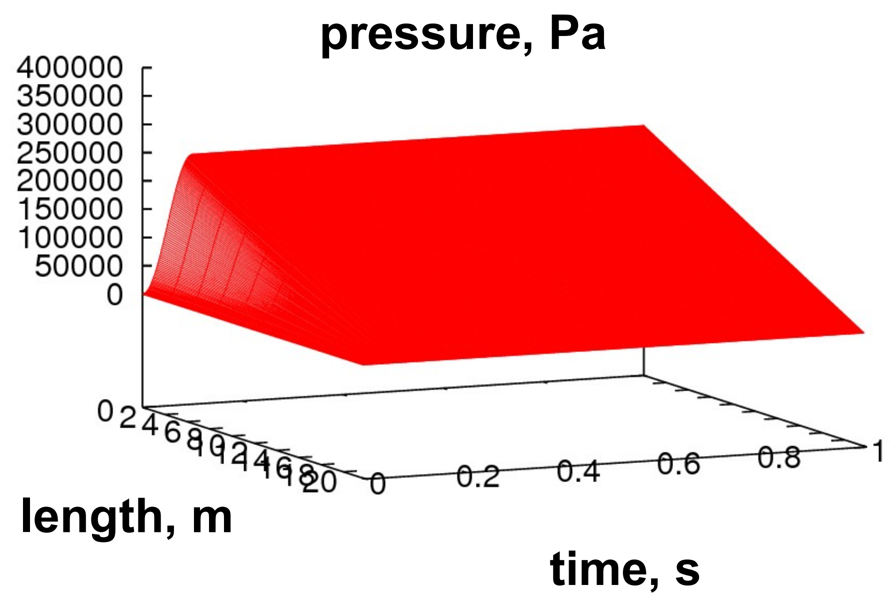


Example: Pressure Wave in Pipeline

MBDyn

- Dynamic pipe (10 “dynamic pipe” elements) compared to static pipe, “smooth” input:

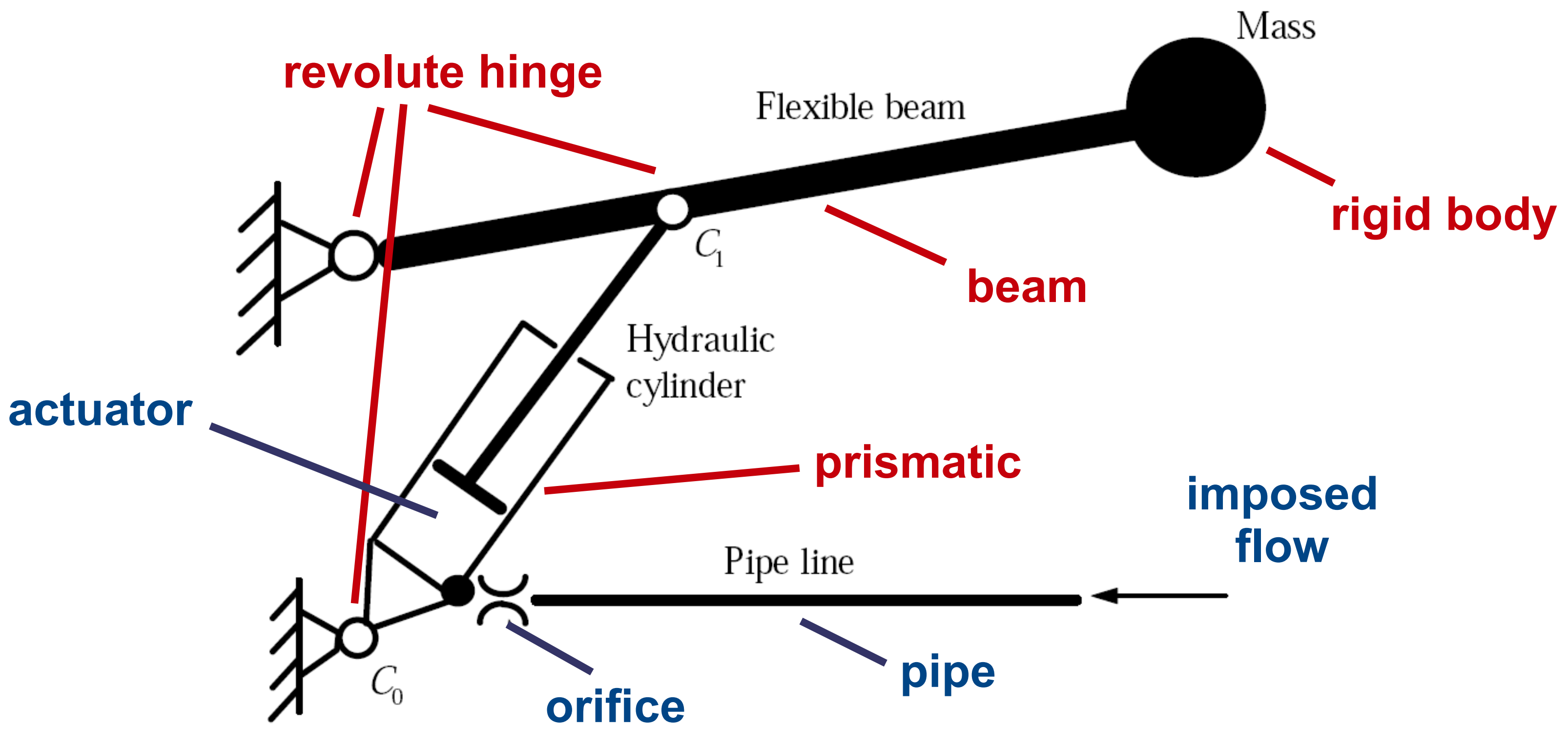




Example: Hydraulically Actuated Beam

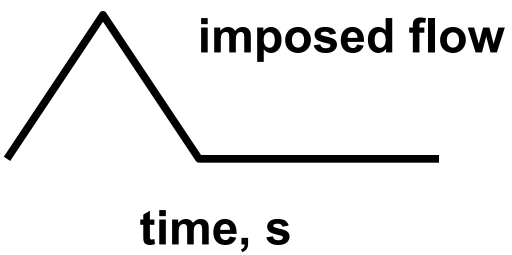
MBDyn

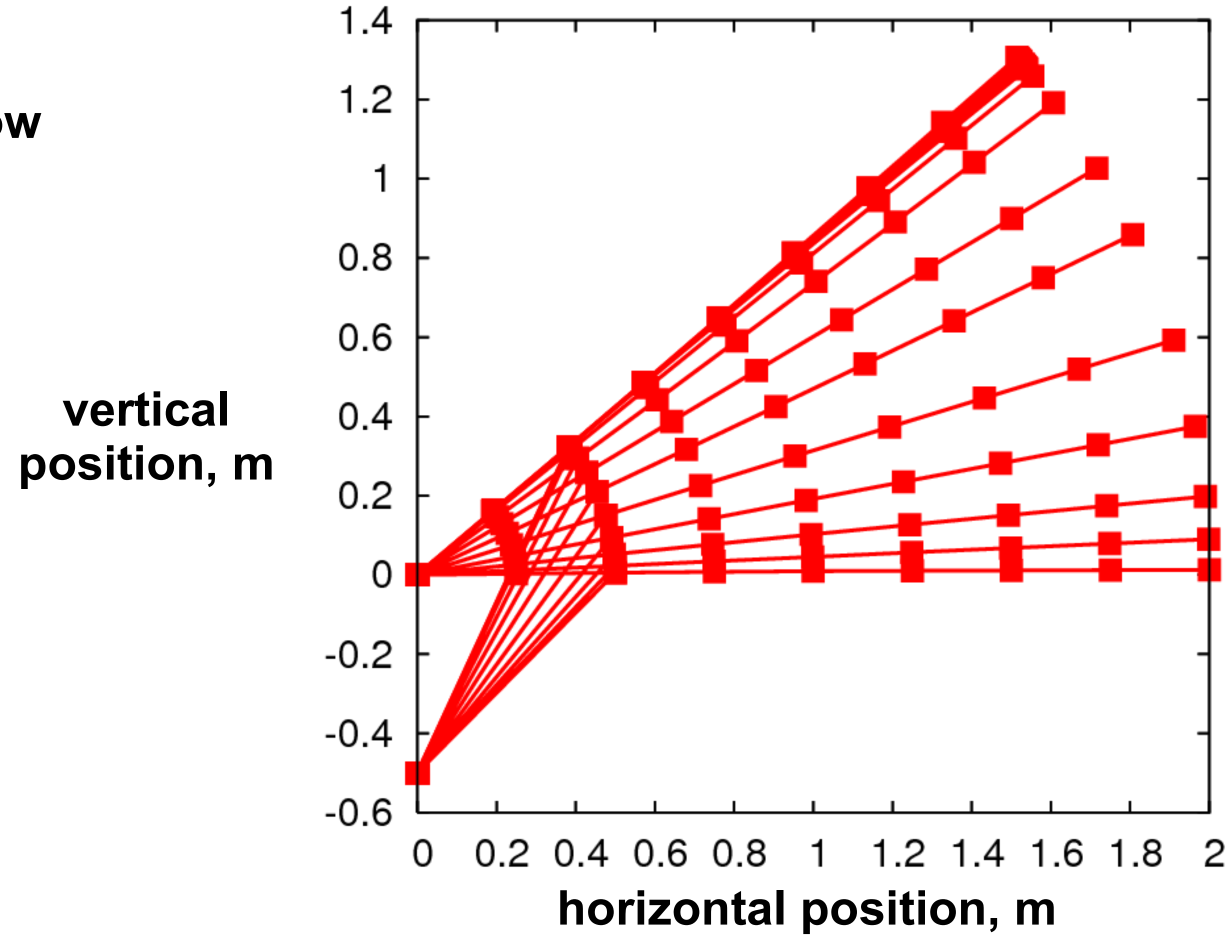
- From: J. Mäkinen, A. Ellman, R. Piché, "Dynamic Simulations of Flexible Hydraulic-Driven Multibody Systems using Finite Strain Beam Theory", 5th Scandinavian International Conference on Fluid Power, Linköping, 1997, Sweden



Example: Hydraulically Actuated Beam

- 4 3-node beam elements

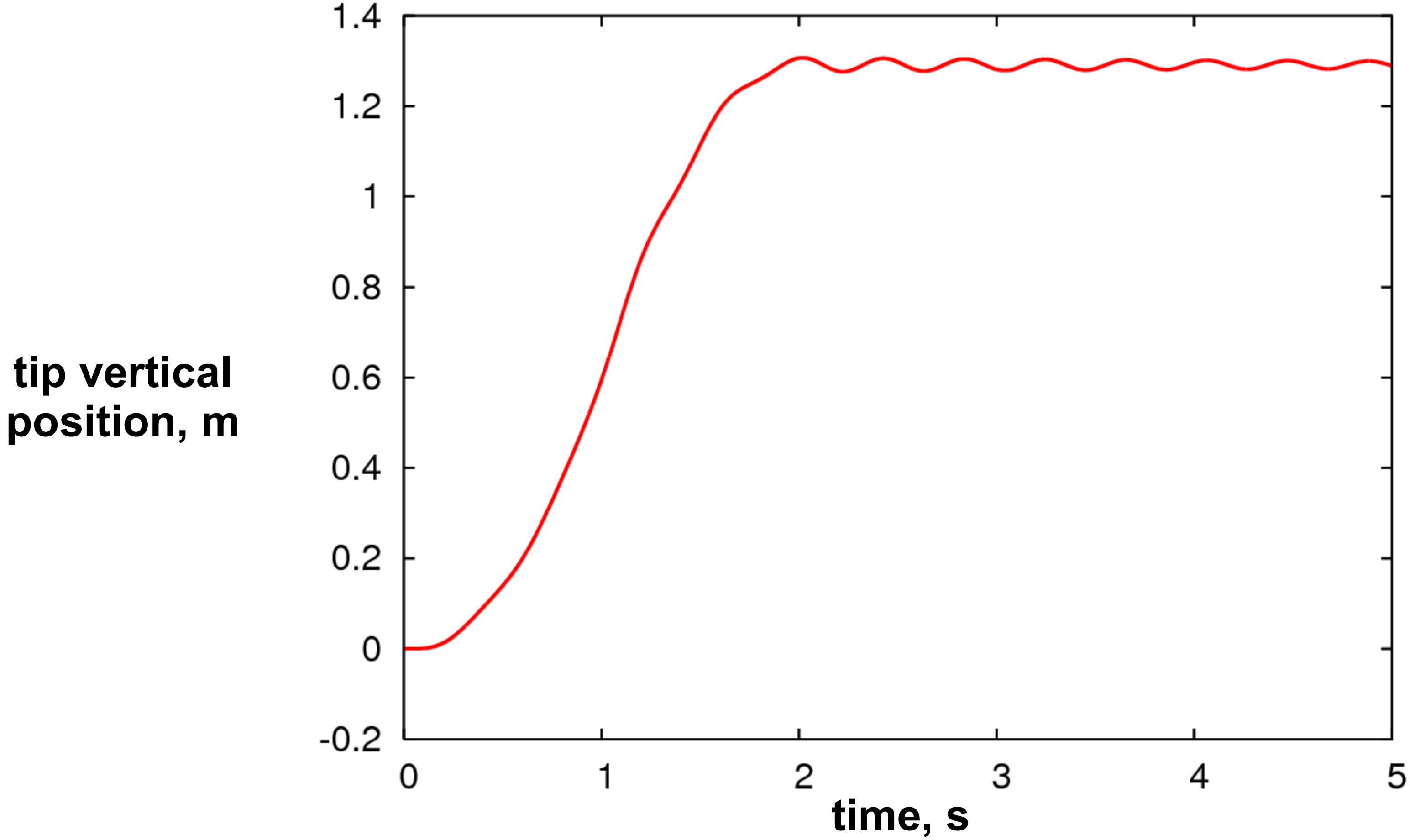




Example: Hydraulically Actuated Beam

MBDyn

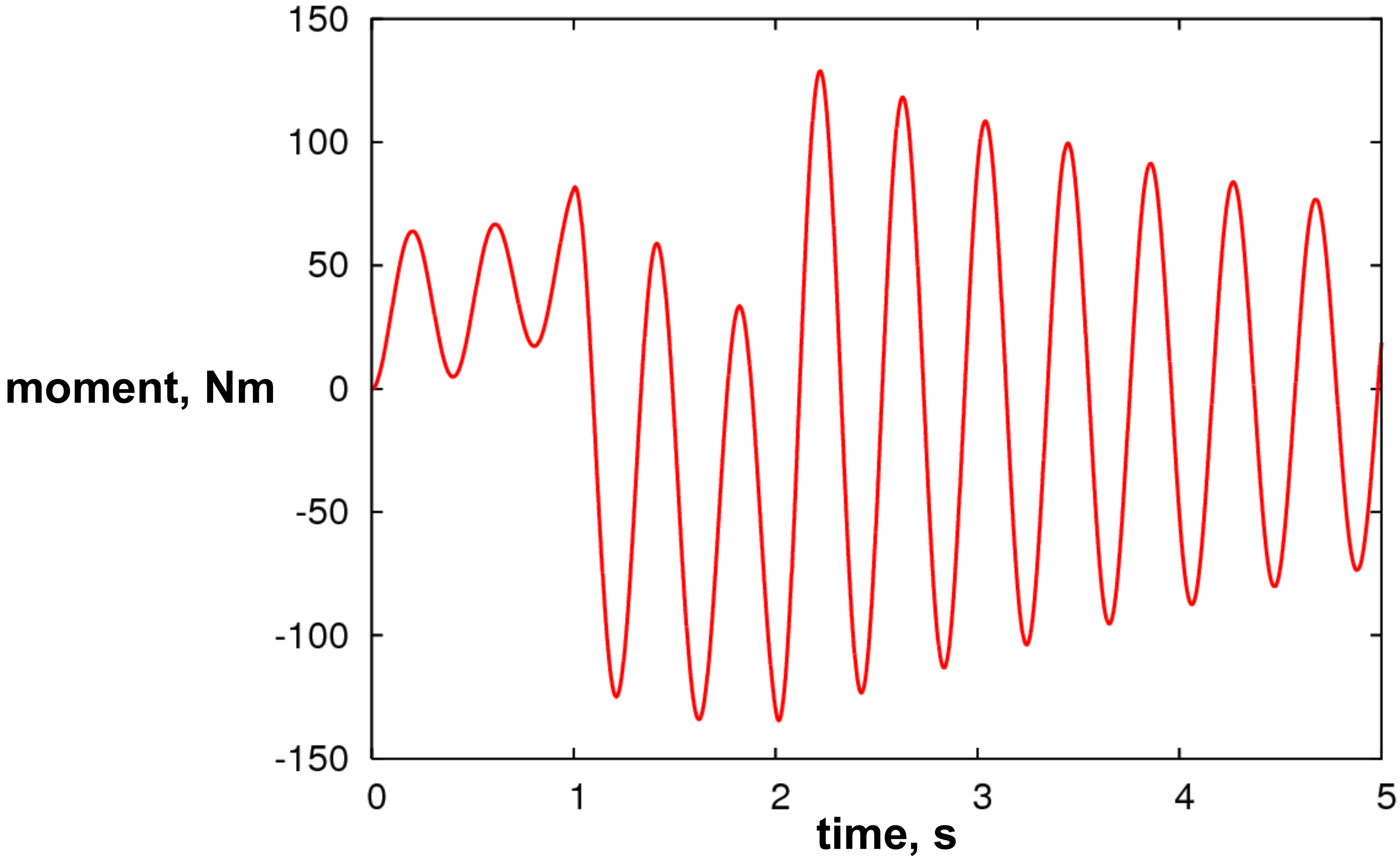
- **Tip vertical displacement**



Example: Hydraulically Actuated Beam

MBDyn

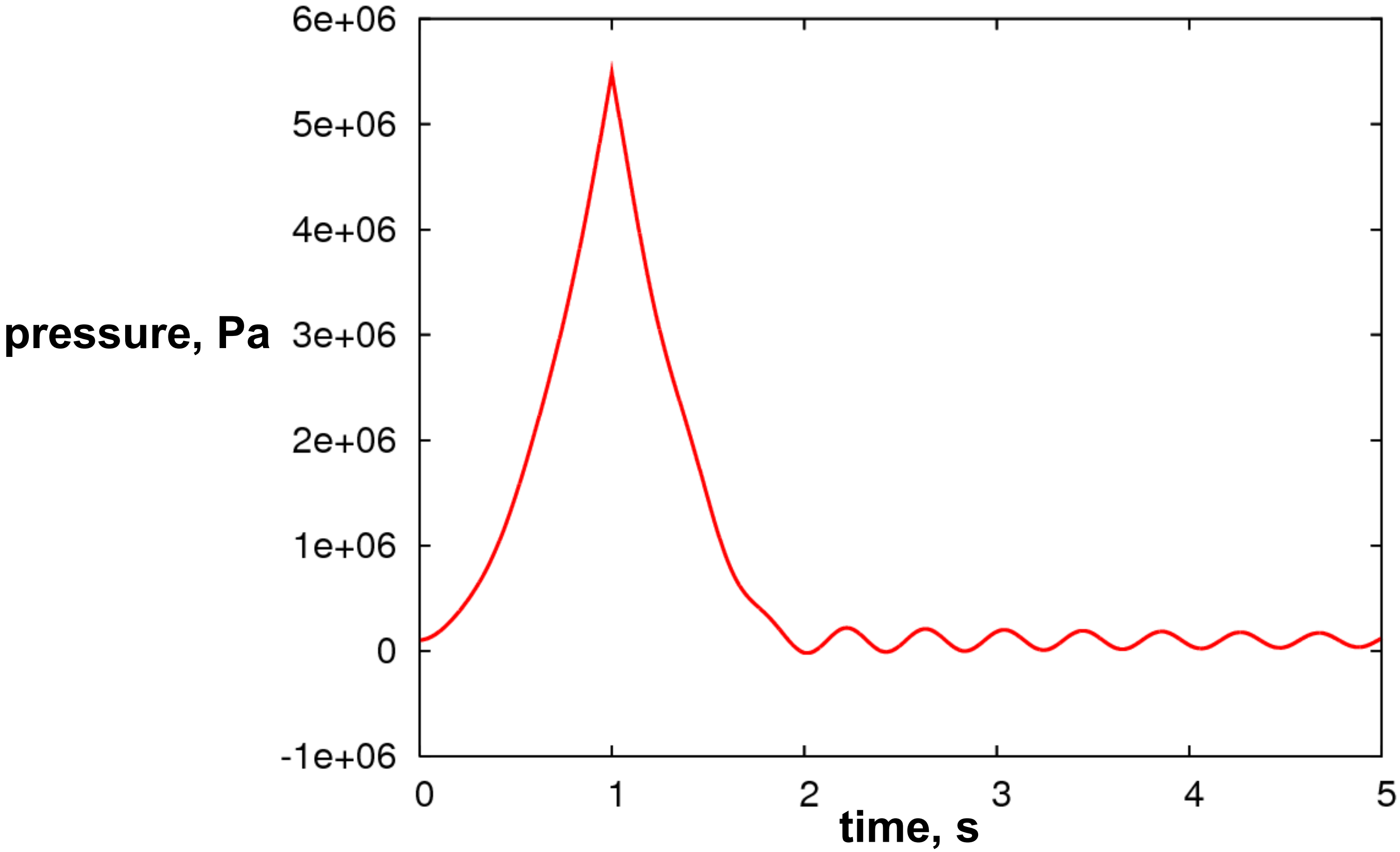
- **Internal moment close to actuator**



Example: Hydraulically Actuated Beam

MBDyn

- **Pressure at imposed flow node**



Consclusions

MBDyn

- **Multibody: natural environment for integration of multiphysics**
- **Need for further development: complete library, build test suite, validate components**
- **The software is free: try it, and feed back!**

mbdyn-users@mbdyn.org

RT-MBDyn

Questions?