Numerical Methods for Rigid Multibody Dynamics

Claus Führer

Centre for Mathematical Sciences
Lund University

Lappennranta 2012

C. Führer Num.Meth. MBS 2012
Unit 0: Preface

- These notes serve as a skeleton for the compact course. They document together with the assignments the course outline and course content.

- All references in the notes refer to the textbook by Eich-Soellner/Führer if not otherwise stated.

- The notes are a guide to read the textbook. They are no textbook.
Unit 1: Equations of Motion

- Unconstrained system: 2nd order ODEs
- Constrained system: 2nd order DAEs
- Linear Implicitness: $O(n)$-methods.
1.1 Multibody System

A multibody system (MBS) consists of:

- **bodies**: mass, inertia, rigid or elastic degrees of freedom (dofs)
- **interconnections**: force elements, e.g., springs, dampers, controllers
- **joints**: constraints which reduce the dofs
1.2 The MBS Simulation Loop

- mechanical system
- multibody model
- establish equations of motion
- solution of equations of motion

\[ M \ddot{p} = f(p, \dot{p}) - G(p)^T \lambda \]
\[ 0 = g(p) \]
1.3 Basic Mathematical Tasks in Multibody Dynamics

- **Kinematic analysis** Given a robot and its hand position. How must the joint angles be chosen? → Nonlinear equation system.

- **Static equilibrium position** How must the forces be chosen such that the system is in equilibrium? → Nonlinear equation system.

- **Dynamic simulation** How does the system behave dynamically? → Numerically Solving ODEs/DAEs.

- **Linear system analysis** Stability? Input/output behavior? → Linearization, Eigenanalysis.
• Design, optimal control How can the behavior of the system be optimized?
  → Parameter Identification, System Optimization, Optimal Control.
1.4 Equations of Motion: Unconstrained System

\[ M \ddot{p} = f_a(t, p, \dot{p}). \]

or in first order format

\[ \begin{align*}
\dot{p} &= v \\
M \dot{v} &= f_a(t, p, v)
\end{align*} \]

\( p \) positions, \( v \) velocities  
\( t \) time  
\( M \) \( n_p \times n_p \) mass matrix  
\( f_a \) applied forces
1.5 Model of Unconstrained Truck
1.6 Coordinates

- \( p_1 \): Vert motion of the rear wheel (body 1)
- \( p_2 \): Vert motion of the truck chassis (body 2)
- \( p_3 \): Rot of the truck chassis (body 2)
- \( p_4 \): Vert motion of the front wheel (body 3)
- \( p_5 \): Vert motion of the driver cabin (body 4)
- \( p_6 \): Rot of the driver cabin (body 4)
- \( p_7 \): Horiz motion of the loading area (body 5)
- \( p_8 \): Vert motion of the loading area (body 5)
- \( p_9 \): Rot of the loading area (body 5)
1.7 Equations of motion: Unconstr. Truck

\[ m_1 \ddot{p}_1 = -f_{102} + f_{122} - m_1 g_{gr} \]
\[ m_2 \ddot{p}_2 = -f_{122} - f_{232} + f_{242} + f_{422} + f_{252} + f_{1d_2} + f_{2d_2} - m_2 g_{gr} \]
\[ l_2 \ddot{p}_3 = (-a_{23} f_{232} - a_{12} f_{122} - h_1 (f_{231} + f_{121})) \cos p_3 - \\
(-a_{23} f_{231} - a_{12} f_{121} - h_1 (f_{232} + f_{122})) \sin p_3 + \\
(a_{25} f_{252} + a_{52} (f_{1d_2} + f_{2d_2}) + h_2 (f_{251} + f_{1d_1} + f_{2d_1})) \cos p_3 - \\
(a_{25} f_{251} + a_{52} (f_{1d_1} + f_{2d_1}) + h_2 (f_{252} + f_{1d_2} + f_{2d_2})) \sin p_3 - \\
(a_{24} f_{242} + a_{42} f_{422} + h_2 (f_{241} + f_{421})) \cos p_3 - \\
(a_{24} f_{241} + a_{42} f_{421} + h_2 (f_{242} + f_{422})) \sin p_3 \]
\[ m_3 \ddot{p}_4 = -f_{302} + f_{232} - m_3 g_{gr} \]
\[ \vdots \]
\[ \vdots \]
1.8 Unconstr. Truck: Geometry and Forces
1.9 Constrained Multibody Systems

\[ M\ddot{p} = f_a(t, p, \dot{p}) - G^T(p)\lambda \]
\[ 0 = g(p) \]

with
\[ g(p) \text{ being } n_c \text{ constraints} \]
\[ G(p) := \frac{d}{dp}g(p) \text{ constraint Jacobian } (n_c \times n_p) \]
\[ \lambda \text{ Lagrange multipliers} \]
1.10 Constrained Truck
1.11 Constrained Truck: Typical Constraint

\[ \rho_{52} = \begin{pmatrix} p_7 \\ p_8 \end{pmatrix} + S(p_9) \begin{pmatrix} -c_{c_1} \\ -c_{c_2} \end{pmatrix} - \begin{pmatrix} 0 \\ p_2 \end{pmatrix} + S(p_3) \begin{pmatrix} -a_{c_1} \\ a_{c_2} \end{pmatrix} = 0. \]
1.12 Different Types of constraints

- holonomic constraints
- non-holonomic constraints
- rhenomic constraints $g(p, t)$
- skleronomic constraints $g(p)$. 

C. Führer Num.Meth. MBS 2012
1.13 3D-Rotation: Kardan Angles

\[ S(\alpha, \beta, \gamma) = S(0, 0, \gamma)S(0, \beta, 0)S(\alpha, 0, 0) \]

with the elementary rotation matrices \( S \), e.g.

\[
S(\alpha, 0, 0) = \begin{pmatrix}
1 & 0 & 0 \\
0 & \cos \alpha & -\sin \alpha \\
0 & \sin \alpha & \cos \alpha
\end{pmatrix}.
\]

\( \alpha \) is the angle of the rotation about the \( x \)-axis, 
\( \beta \) the angle of rotation about the (new) \( y \)-axis 
and \( \gamma \) the angle about the (even newer) \( z \)-axis.

C. Führer Num.Meth. MBS 2012
1.14 3D-Rotation: angular velocity

Poisson equation:

\[
\dot{S}(\alpha, \beta, \gamma) = \tilde{\omega} S(\alpha, \beta, \gamma).
\]

with \( \tilde{\omega} \):

\[
\tilde{\omega} := \begin{pmatrix}
0 & -\omega_3 & \omega_2 \\
\omega_3 & 0 & -\omega_1 \\
-\omega_2 & \omega_1 & 0
\end{pmatrix}
\]

This defines the angular velocities \( \omega_1, \omega_2, \omega_3 \).
1.15 3D-Rotation: angular velocity vs $\dot{p}$

\[
\begin{pmatrix}
\dot{\alpha} \\
\dot{\beta} \\
\dot{\gamma}
\end{pmatrix} =
\begin{pmatrix}
\cos \gamma / \cos \beta & \sin \gamma / \cos \beta & 0 \\
\sin \gamma & -\cos \gamma & 0 \\
-\cos \gamma \tan \beta & -\sin \gamma \tan \beta & 1
\end{pmatrix}
\begin{pmatrix}
\omega_1 \\
\omega_2 \\
\omega_3
\end{pmatrix}.
\]

**Singularities !**

C. Führer Num.Meth. MBS 2012
1.16 3D-Rotation: complete system

\[ \dot{p} = Z(p)v \]
\[ M\dot{v} = f_a(t, p, v, s) - Z(p)^T G(p)^T \lambda \]
\[ 0 = g(p) \]

(see also Exercise 1).
1.17 Relative vs Absolute coordinates

Absolute coordinates

• Mass matrix constant and block-diagonal.

• Constraints $\rightarrow$ DAEs.

Relative coordinates:

• Mass matrix dense and position dependent.

• For tree structured systems the constraints can be directly eliminated.
1.18 Tree structured systems
1.19 Truck: relative coordinates

\[ q_1 = \|\rho_{10}\| \]
\[ q_2 = \|\rho_{12}\| \]
\[ q_3 = p_3 \]
\[ q_4 = \|\rho_{23}\| \]
\[ q_5 = \|\rho_{24}\| \]
\[ q_6 = p_6 - p_3 \]
\[ q_7 = p_9 - p_3 \cdot \]

7 degrees of freedom (dofs)
1.20 Relation between absolute/relative positions

Absolute $\rightarrow$ relative: $n_q = n_{dof}$ equations

$$0 = \tilde{g}(p, q),$$

Relative $\rightarrow$ absolute: $n_p = n_c + n_q$ equations

$$0 = \begin{pmatrix} g(p) \\ \tilde{g}(p, q) \end{pmatrix} =: \gamma(p, q).$$
1.21 Relation between absolute/relative velocities

Derivative assumed to be regular (Grübler condition):

\[ \Gamma_p(p, q) = \frac{\partial}{\partial p} \gamma(p, q) \]

Velocities:

\[ 0 = \frac{\partial}{\partial p} \gamma(p, q) \dot{p} + \frac{\partial}{\partial q} \gamma(p, q) \dot{q}, \]

\[ =: \Gamma_p(p,q) \]

and

\[ \dot{p} = -\Gamma_p(p,q)^{-1}\Gamma_q(p,q) \dot{q}, \]

\[ =: V(p,q) \]
1.22 Relation between absolute/relative accelerations

\[ \ddot{p} = V(p, q)\dot{q} + \zeta(p, q, \dot{p}, \dot{q}). \]

with \( \zeta \) collecting all terms with lower derivatives of \( p \) and \( q \).
1.23 State Space Form in relative coordinates

We note

\[ G(p)V(p, q) = 0. \]

From this we obtain

\[ \ddot{q} = V(p, q)\left( f_a(p, \dot{p}) - M \zeta(p, q, \dot{p}, \dot{q}) \right). \]

Note: Mass matrix looses its structure and becomes state dependent.

C. Führer Num.Meth. MBS 2012
1.24 Mixed Coordinate Formulation for Tree Structured Systems

For chain like structures, mixed coordinate formulations are much more efficient.

→ block diagonal matrices to invert.

C. Führer Num.Meth. MBS 2012
1.25 Mixed Coordinate Formulation for Tree Structured Systems

\[
\begin{pmatrix}
M & 0 \\
0 & 0
\end{pmatrix}
\begin{pmatrix}
\ddot{p} \\
\ddot{q}
\end{pmatrix}
= \begin{pmatrix}
\dot{f}_a(t, p, \dot{p}) \\
0
\end{pmatrix}
- \begin{pmatrix}
\Gamma_p(p, q)^T \\
\Gamma_q(p, q)^T
\end{pmatrix} \mu
\]

\[0 = \gamma(p, q).\]

We note the \(i\)-th interconnection has the form

\[\gamma^i(p, q) = \gamma^i(p^i, p^{i-1}, q^i).\]

Resorting of components gives ...
Resorting: \(x = (\ldots, (\dot{p}^i)^T, (\dot{q}^i)^T, (\mu^i)^T, (\dot{p}^{i-1})^T, (\dot{q}^{i-1})^T, (\mu^{i-1})^T, \ldots)^T\)

\[
\begin{pmatrix}
A_k & C_k^T \\
C_k & A_{k-1} & C_{k-1}^T \\
C_{k-1} & A_{k-2} & C_{k-2}^T \\
C_{k-2} & & & \ddots \\
C_3 & A_2 & C_2^T \\
C_2 & A_1 & C_1^T \\
C_1 & A_0 \\
\end{pmatrix}
\begin{pmatrix}
x^k \\
x^{k-1} \\
x^{k-2} \\
\vdots \\
x^2 \\
x^1 \\
x^0 \\
\end{pmatrix}
=
\begin{pmatrix}
b^k \\
b^{k-1} \\
b^{k-2} \\
\vdots \\
b^2 \\
b^1 \\
b^0 \\
\end{pmatrix}
\]

with \(x^i = ((\dot{p}^i)^T, (\dot{q}^i)^T, (\mu^i)^T)^T\) and \(b^i = ((f^i)^T, 0, (z^i)^T)^T\).
1.27 Mixed Coordinate Formulation for ... (Cont.)

\[ A_i := \begin{pmatrix} M_i & 0 & \Gamma^T_{p,i,i} \\ 0 & 0 & \Gamma^T_{q,i,i} \\ \Gamma_{p,i,i} & \Gamma_{q,i,i} & 0 \end{pmatrix}. \]

(1)

\( A_i \) maximal dimension \( 18 = 3 \times 6 \).

\[ C^T_i := \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \Gamma_{p,i-1} & 0 & 0 \end{pmatrix}. \]
a.) Downward recursion

- Initialization:
  \[ \hat{A}_k := A_k \quad \hat{b}^k := b^k. \]

- Recursion (i=k-1:-1:0)
  \[ \begin{align*}
  \hat{A}_i & := A_i - C_{i+1} \hat{A}_{i+1}^{-1} C_{i+1}^T \\
  \hat{b}^i & := b^i - C_i \hat{A}_i^{-1} \hat{b}^{i+1}.
  \end{align*} \]

b.) Upward recursion:

Solve for \( x^0 \) and then for \( x^i \):

\[ \begin{align*}
\hat{A}_0 x^0 &= \hat{b}^0 \\
\hat{A}_i x^i &= \hat{b}^i - C_i^T x^{i-1} \text{ with } i = 1 : k.
\]
1.29 Linearization

Some assumptions

\[ f_a(t, p, v) = f_a(p, v, u(t)) \]

\( u(t) \) plays in the sequel the role of a given input function

Time dependency of the constraint = kinematic excitation

\[ g(t, p) = g(p) - z(t). \]
1.30 Linearization: Nominal Solution

Nominal Solutions

\[ p_N(t), v_N(t) \text{ and } \lambda_N(t) \]

and

Nominal input

\[ u_N(t) = 0 \text{ and } z_N(t) = 0 \]
1.31 Linearization: Taylor

\[ \Delta \dot{p} = \Delta v \]
\[ M(t) \Delta \dot{v} = -K(t) \Delta p - D(t) \Delta v - G(t) \Delta \lambda + B(t) u(t) \]
\[ 0 = G'(t) \Delta p - z(t) \]

Here: Numerical Linearization, see Lab 1

C. Führer Num.Meth. MBS 2012
1.32 Linearization: Matrices

mass matrix \[ M(t) := M(p_N(t)) \]

stiffness matrix \[ K(t) := (\dot{v}_N^T \frac{\partial}{\partial p} M(p) - \frac{\partial}{\partial p} f_a(p, v_N(t), 0) + \frac{\partial}{\partial p} G(p)^T \lambda_N)_{p=p_N(t)} \]

damping matrix \[ D(t) := -\left(\frac{\partial}{\partial v} f_a(p_N(t), v, 0)\right)_{v=v_N(t)} \]

constraint matrix \[ G(t) := \left(\frac{d}{dp} g(p)\right)_{p=p_N(t)} \]

input matrix \[ B(t) := \left(\frac{\partial}{\partial u} f_a(p_N(t), v_N(t), u)\right)_{u=0} \]