REAL-TIME METHODS IN FLEXIBLE MULTIBODY DYNAMICS

A thesis submitted for the degree of
Doctor Ingeniero Industrial

Urbano Lugarís Armesto

University of La Coruña

Ferrol, November 2008
Outline

Introduction

Formulation in Relative Coordinates

Inertia Shape Integrals

Geometric Stiffening

Conclusions and Future Research
Outline

Introduction

Formulation in Relative Coordinates

Inertia Shape Integrals

Geometric Stiffening

Conclusions and Future Research
Motivation (I)

- Our group has developed very efficient and robust formulations for the real–time simulation of rigid multibody systems.
Objective: include flexibility in real–time applications
- Simulators, virtual reality...

Many multibody applications cannot neglect flexibility
- Slender components
- Newer lightweight materials
- High operational speed

Flexible bodies require a higher computational effort
- Elastic forces
- Variable mass matrix
Existing Flexible MBS Approaches

- **Inertial frame**
  - One inertial frame common to all the bodies in the system
  - J.C. Simó, L. Vu–Quoc, A. Cardona, M. Géradin, A.A. Shabana

- **Floating frame: most efficient**
  - One reference frame attached to each flexible body
  - E.J. Haug, A.A. Shabana, R.A. Wehage

- **Corotational frame**
  - Each finite element has a local frame of reference
  - T. Belytschko, B.J. Hsieh
Reference Coordinates ≡ Rigid Body Coordinates

- Reference point coordinates
  - Position and orientation in Cartesian coordinates

- Natural coordinates
  - Fully Cartesian
  - Points and unit vectors

- Relative coordinates
  - \( O(n) \) fully–recursive formulations
  - \( O(n^3) \) semi–recursive formulations
Objectives and Scope of the Present Work

- Development of a semi-recursive $O(n^3)$ FFR formulation
  - Based on an existing rigid-body one
- Comparison between natural and relative coordinates
  - Same comparison has been previously carried out in the rigid case
  - FFR formulation in natural coordinates as a reference
  - Both formulations share the same flexible body modeling
- Optimization of the inertia terms
  - Inertia Shape Integrals preprocessing
  - Implement in both absolute and relative coordinates
- Extension to nonlinear problems
  - Implementation and comparison of three techniques for capturing geometric stiffening in beams
  - Substructuring, Nonlinear stiffness matrix and Foreshortening
Outline

Introduction

Formulation in Relative Coordinates

Inertia Shape Integrals

Geometric Stiffening

Conclusions and Future Research
General flexible body

- Position of a point: \( \mathbf{r} = \mathbf{r}_0 + \mathbf{A} (\mathbf{r}_u + \mathbf{r}_f) \); \( \mathbf{A} = [u \ v \ w] \)
- Elastic displacement in local coordinates (Craig–Bampton)

\[
\mathbf{r}_f = \sum_{i=1}^{n_s} \Phi_i \eta_i + \sum_{j=1}^{n_d} \Psi_j \xi_j = \mathbf{X}\mathbf{y}
\]
Craig–Bampton Reduction

Static modes: unit displacements at boundaries

Dynamic modes: normal eigenmodes with fixed boundaries
Recursive Kinematics

- Closed loops are cut $\implies$ constraints $\Phi$
- Dependent relative coordinates $\mathbf{z}$
- Intermediate dynamic terms in Cartesian coordinates: $\mathbf{M}, \mathbf{Q}$
- Cartesian coordinates defined at velocity level (reference)

$$\mathbf{Z}^T = \{\mathbf{s}^T, \mathbf{\omega}^T, \mathbf{\dot{y}}^T\}$$

- Recursive relationships for velocities and accelerations

$$\mathbf{Z}_{rj} = \mathbf{Z}_{ri} + b_j \mathbf{\dot{z}}_j$$
$$\mathbf{\dot{Z}}_{rj} = \mathbf{\dot{Z}}_{ri} + b_j \mathbf{\ddot{z}}_j + \mathbf{d}_j$$

$$\implies \mathbf{Z} = \mathbf{R} \mathbf{\dot{z}}$$
Projection of the Dynamic Terms

- Static modes behave analogously as kinematic joints

- Kinematic relations include now joints and static modes

\[
\begin{align*}
Z_{rj} &= Z_{ri} + b_j \dot{z}_j + \varphi_j^P \dot{\eta}_j^P \\
\dot{Z}_{rj} &= \dot{Z}_{ri} + b_j \ddot{z}_j + \varphi_j^P \ddot{\eta}_j^P + d_j + \gamma_j^P
\end{align*}
\]

- Projection into \( z \):

\[
M = R^T \tilde{M} R; \quad Q = R^T \left( \tilde{Q} - \tilde{M} \dot{R} \dot{z} \right)
\]
Calculation of the Inertia Terms

- Corotational approximation

\[
T = \frac{1}{2} \int_V \dot{\mathbf{r}}^T \dot{\mathbf{r}} \, dm = \frac{1}{2} \dot{\mathbf{r}}^*^T \left( \int_V \mathbf{N}^T \mathbf{N} \, dm \right) \dot{\mathbf{r}}^* = \frac{1}{2} \dot{\mathbf{r}}^*^T \mathbf{M}^* \dot{\mathbf{r}}^*
\]

- Transformation matrix \( \mathbf{B} \), assembled for the whole body

\[
\mathbf{B}^* = \begin{bmatrix}
\mathbf{B}_1 \\
\mathbf{B}_2 \\
\vdots \\
\mathbf{B}_n \\
\end{bmatrix} = \begin{bmatrix}
\mathbf{I}_3 & -\tilde{\mathbf{r}}_1 & \mathbf{A}X_1 \\
\mathbf{I}_3 & -\tilde{\mathbf{r}}_2 & \mathbf{A}X_2 \\
\vdots & \vdots & \vdots \\
\mathbf{I}_3 & -\tilde{\mathbf{r}}_n & \mathbf{A}X_n \\
\end{bmatrix} \implies \dot{\mathbf{r}}^* = \mathbf{B}^* \mathbf{Z}
\]

- Projection of the finite element mass matrix:

\[
\bar{\mathbf{M}} = \mathbf{B}^{*T} \mathbf{M}^* \mathbf{B}^*
\]

- Velocity–dependent inertia forces:

\[
\bar{\mathbf{Q}_v} = -\mathbf{B}^{*T} \mathbf{M}^* \mathbf{B}^* \mathbf{Z}
\]
Assembly of the Equations of Motion

Relative positions and velocities $z, \dot{z}$

FORWARD RECURSIVE ANALYSIS
- Natural coordinates $q_i, q_i$
  - Kinematic terms in $Z b_i, d_i, \phi_i^p, \gamma_i^p$
  - Dynamic terms in $Z \mathbf{M}_i, \mathbf{Q}_i$

BACKWARD ACCUMULATION
- Projection of the dynamic terms $\mathbf{M}, \mathbf{Q}$
- Constraints and Jacobian $\Phi, \Phi_z$

EQUATIONS OF MOTION
$$\mathbf{M}\ddot{z} + \Phi_z^T \alpha \Phi + \Phi_z^T \lambda^* = \mathbf{Q}$$
Dynamic Formulation and Numerical Integration

- **Index–3 Augmented Lagrangian**

\[
M \ddot{z} + \Phi_z^T \alpha \Phi + \Phi_z^T \lambda^* = Q \\
\lambda_{i+1} = \lambda_i^* + \alpha \Phi \quad i = 1, 2, \ldots
\]

- **Newmark integrator**

\[
\ddot{z}_{n+1} = f \left(z_{n+1}, z_n, \dot{z}_n, \ddot{z}_n\right) \\
\dddot{z}_{n+1} = f \left(z_{n+1}, z_n, \dot{z}_n, \ddot{z}_n\right)
\]

- **Combination of formulation and integrator: Newton–Raphson**

\[
f_z \approx M + \gamma h C + \beta h^2 \left(\Phi_z^T \alpha \Phi_z + K\right) \\
f = \beta h^2 \left(M \ddot{q} + \Phi_z^T \alpha \Phi + \Phi_z^T \lambda^* - Q\right)
\]

- **Velocity and acceleration projections**

\[
f_z \dot{z} = W \dot{z}^* - \beta h^2 \Phi_z^T \alpha \Phi_t \\
f_z \ddot{z} = W \ddot{z}^* - \beta h^2 \Phi_z^T \alpha \left(\dot{\Phi}_z \dot{z} + \dot{\Phi}_t\right)
\]
First Example: 2D Double Four–Bar Mechanism

- Five 1 Kg, 1 m long steel bars
- All bars can be flexible or not
- 2 static modes and 2 dynamic modes per bar
- 1 m/s initial velocity, gravity
- Integration: 5 s (2.5 turns)

<table>
<thead>
<tr>
<th># flexible bars</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute</td>
<td>6</td>
<td>13</td>
<td>20</td>
<td>27</td>
<td>34</td>
<td>41</td>
</tr>
<tr>
<td>Relative</td>
<td>5</td>
<td>8</td>
<td>11</td>
<td>14</td>
<td>17</td>
<td>20</td>
</tr>
</tbody>
</table>
First Example: 2D Double Four–Bar Mechanism

- Five 1 Kg, 1 m long steel bars
- All bars can be flexible or not
- 2 static modes and 2 dynamic modes per bar
- 1 m/s initial velocity, gravity
- Integration: 5 s (2.5 turns)

<table>
<thead>
<tr>
<th># flexible bars</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute</td>
<td>0.91</td>
<td>3.30</td>
<td>6.24</td>
<td>9.61</td>
<td>11.51</td>
<td>15.22</td>
</tr>
<tr>
<td>Relative</td>
<td>4.85</td>
<td>9.11</td>
<td>12.62</td>
<td>15.74</td>
<td>17.74</td>
<td>20.92</td>
</tr>
</tbody>
</table>
Second Example: Iltis Suspension

- Up to three flexible bodies
- Structural damping added
- Initial equilibrium position
- Runs down 20 cm step
- Motion integrated along 5 s
- Implemented in FORTRAN

![Diagram of Iltis Suspension]

Graph showing the number of variables for different numbers of flexible bodies:

- None: 35
- 1: 47
- 2: 60
- 3: 72

Legend:
- Absolute
- Relative

Number of flexible bodies:
- None
- 1
- 2
- 3

Number of variables:
- 10
- 20
- 30
- 40
- 50
- 60
- 70
- 80
Results: Iltis Suspension

Time histories in the vertical direction

- Chassis height (m)
- Wheel center height (m)

Time (s)
Results: Iltis Suspension

CPU–time vs. number of flexible bodies

<table>
<thead>
<tr>
<th>Number of flexible bodies</th>
<th>Absolute</th>
<th>Relative</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.15</td>
<td>0.2</td>
</tr>
<tr>
<td>3</td>
<td>0.3</td>
<td>0.3</td>
</tr>
</tbody>
</table>
Third Example: Full Vehicle

- Iltis vehicle: 4 suspensions
- Structural damping added
- Initial velocity: 5 m/s
- Road profile: ramp + steps
- Motion integrated along 8 s
- Implemented in FORTRAN

![Graph showing number of variables and flexible bodies](image)

<table>
<thead>
<tr>
<th>Number of flexible bodies</th>
<th>Absolute</th>
<th>Relative</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>168</td>
<td>34</td>
</tr>
<tr>
<td>4</td>
<td>212</td>
<td>56</td>
</tr>
<tr>
<td>8</td>
<td>258</td>
<td>76</td>
</tr>
<tr>
<td>12</td>
<td>304</td>
<td>98</td>
</tr>
</tbody>
</table>
Results: Full Vehicle

Time history in vertical direction: chassis
Time history in vertical direction: center of front left wheel
Results: Full Vehicle

Time history of the deflection of the A–arm

Tip deflection (mm)

Time (s)
Results: Full Vehicle

CPU–time vs. number of flexible bodies

- Absolute
- Relative

<table>
<thead>
<tr>
<th>Number of flexible bodies</th>
<th>CPU-time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>1.0</td>
</tr>
<tr>
<td>4</td>
<td>2.0</td>
</tr>
<tr>
<td>8</td>
<td>3.0</td>
</tr>
<tr>
<td>12</td>
<td>4.0</td>
</tr>
</tbody>
</table>
Conclusions of the Second Chapter

- New semi-recursive $O(n^3)$ FFR formulation successfully implemented and tested

- Very good correlation of results between both formulations

- Results in the flexible case similar to the rigid case
  - Absolute coordinates are faster in small systems
  - Relative coordinates are more efficient for large systems

- Common bottleneck: the inertia terms
  - Projection of finite element mass matrix is time-consuming
  - Solution addressed in the third chapter: Inertia Shape Integrals
Outline

Introduction

Formulation in Relative Coordinates

Inertia Shape Integrals

Geometric Stiffening

Conclusions and Future Research
Background

- Starting point: two efficient FFR formulations
  - Method in absolute (natural) coordinates
  - Method in relative coordinates
- Finite element model reduced from order $N$ to order $n$
- Variable inertia terms obtained by order $N$ velocity projections
- Bottleneck: projections take up to 80% CPU-time

Solution: preprocessing (inertia shape integrals)
- Order $n$ matrix operations at every time-step
Preprocessing Approach

- The mass matrix can be directly obtained by integrating $B^T B$

$$T = \frac{1}{2} \int_V \dot{\mathbf{r}}^T \dot{\mathbf{r}} \, dm = \frac{1}{2} \mathbf{q}^T \left( \int_V B^T B \, dm \right) \dot{\mathbf{q}} \implies \mathbf{M} = \int_V B^T B \, dm$$

- Different integrals are needed depending on the formulation
  - Absolute: $\mathbf{M} = \int_V \begin{bmatrix} I_3 & \tilde{r}_1 I_3 & \tilde{r}_2 I_3 & \tilde{r}_3 I_3 & \mathbf{AX} \\ \tilde{r}_1^2 I_3 & \tilde{r}_1 \tilde{r}_2 I_3 & \tilde{r}_1 \tilde{r}_3 I_3 & \tilde{r}_1 \mathbf{AX} & \tilde{r}_1 \mathbf{AX} \\ \tilde{r}_2^2 I_3 & \tilde{r}_2 \tilde{r}_3 I_3 & \tilde{r}_2 \mathbf{AX} & \tilde{r}_2 \mathbf{AX} & \mathbf{X}^T \mathbf{X} \\ \mathbf{sym.} & \mathbf{sym.} & \mathbf{sym.} & \mathbf{sym.} & \mathbf{X}^T \mathbf{X} \end{bmatrix} \, dm$

  - Relative: $\mathbf{\tilde{M}} = \int_V \begin{bmatrix} I_3 & -\tilde{\mathbf{r}} & \mathbf{AX} \\ -\tilde{\mathbf{r}}^T & \tilde{\mathbf{r}}^T & \tilde{\mathbf{r}} \mathbf{AX} \\ \mathbf{sym.} & \mathbf{sym.} & \mathbf{X}^T \mathbf{X} \end{bmatrix} \, dm$

- Centrifugal and Coriolis forces are obtained as $-\int_V B^T \dot{\mathbf{B}} \dot{\mathbf{q}} \, dm$
Inertia Shape Integrals

- 13 constant integrals, including scalars, vectors and matrices
  - Mass, undeformed static moment and planar inertia tensor
    \[ m = \int_V dm; \quad \tilde{m}_u = \int_V \tilde{r}_u dm; \quad \tilde{P}_u = \int_V \tilde{r}_u \tilde{r}_u^T dm \]

- Four $3 \times n$ matrices
  \[ S = \int_V X dm; \quad S^i = \int_V \tilde{r}_{ui} X dm; \quad i = 1, 2, 3 \]

- Six $n \times n$ matrices
  \[ S^{ij} = \int_V X_i^T X_j dm; \quad i, j = 1, 2, 3 \]
CPU–time vs. Finite Element Mesh Size

Absolute coordinates

Relative coordinates

Projection

Preprocessing

CPU-time (s)

Elements per bar
Conclusions of the Third Chapter

- Preprocessing using inertia shape integrals has been implemented in both the absolute and the relative formulations.

- The formulation in relative coordinates keeps its advantage over the absolute one for large size problems.

- The use of preprocessing always improves efficiency:
  - Improvement obtained even for small finite element models.
  - Preprocessing time has no significant impact.
  - The $B$ matrix method is easier to implement.

- Small models ($< 10$ finite elements): $B$ matrix
- Large models ($> 20$ finite elements): inertia shape integrals
Outline

Introduction

Formulation in Relative Coordinates

Inertia Shape Integrals

Geometric Stiffening

Conclusions and Future Research
Background

- Geometric stiffening appears in rotating beams, such as helicopter or turbine blades, increasing bending stiffness with rotation speed.
- When FFR formulations are used, this effect can be lost if linear elastic displacements are assumed in the FE model.
Objectives

- Objective: extend the range of usability of FFR formulations by including geometric stiffening

- Three different techniques are studied
  - Substructuring
  - Nonlinear stiffness matrix
  - Foreshortening

- These techniques are implemented and compared in absolute and relative coordinates (substructuring only in relative coordinates)
Substructuring

- The beam is divided into several substructures:

- Each substructure is a standard FFR flexible body
- Substructures are interconnected by *bracket joints*
- Most general approach, the FFR formulation is not modified
- Tested only in relative coordinates
  - Natural coordinates: $12 + n_m$ variables per substructure
  - Relative coordinates: $n_m$ variables per substructure
Nonlinear Stiffness Matrix: Potential Energy

- Strain energy of an Euler–Bernoulli beam

\[
U = \frac{1}{2} \int_0^L E A u'^2 \, dx + \frac{1}{2} \int_0^L E I v''^2 \, dx
\]

**Linear formulation**

\[
+ \frac{1}{2} \int_0^L E A u'_0 v'_0^2 \, dx + \frac{1}{8} \int_0^L E A v'_0^4 \, dx
\]

**First nonlinear**

- Introduces coupling between axial and transversal displacement

**Second nonlinear**

- Linear formulation: only the first two terms are retained
  - Axial and transversal displacements are independent
- First nonlinear formulation: the third term is added
- Second nonlinear formulation: full strain energy expression
Nonlinear Stiffness Matrix: Elastic Forces

- Linear formulation
  - Constant stiffness matrix
  - No coupling between axial and transversal stiffness
  \[ F_{el} = -K_L y \]

- First nonlinear formulation
  - Variable stiffness matrix
  - \( K_G \) couples axial and transversal stiffness
  \[ F_{el} = -(K_L + K_G) y; \quad K_G = \sum_{i=1}^{ns} \eta_i K_{Gi} + \sum_{j=1}^{nd} \xi_j K_{Gj} \]

- Second nonlinear formulation
  - Highly nonlinear stiffness matrix
  \[ F_{el} = -(K_L + K_G + K_H) y + Q_G \]
Foreshortening

- Foreshortening: axial shortening produced by deflection

- Modified axial displacement

\[ u_0 = s + u_{fs}; \quad u_{fs}(x) = -\frac{1}{2} \int_{x_0}^{x} v'_0 \, dx \]

- Strain energy equivalent to second nonlinear formulation
- Introduced in the axial components of the mode shapes \( \mathbf{X} \)
  - It renders the \( \mathbf{X} \) matrix variable \( \Rightarrow \mathbf{B}^* \) projection
  - Linear elastic forces with unmodified \( \mathbf{K}_L \) matrix
  - Geometric stiffening is introduced at the kinematics level
- Captures the effect with no axial modes
System Under Test: Rotating Beam

- Steel beam pinned at one end

- Guided rotation about the origin

\[
\omega(t) = \begin{cases}
\frac{\Omega_s}{T_s} \left[ t - \left( \frac{T_s}{2\pi} \right) \sin \left( \frac{2\pi t}{T_s} \right) \right] & 0 \leq t < T_s \\
T_s \leq t & \text{otherwise}
\end{cases}
\]

- 2D and 3D cases studied; motion is integrated along 20 s
- Deflection at the tip is measured for $\Omega_s = 6$ rad/s, $T_s = 15$ s
- Results are compared to a reference solution (ANCF or FEM)
Results: Horizontal Deflection in the 2D Case

Linear formulation

![Graph showing horizontal deflection over time]

<table>
<thead>
<tr>
<th>Method</th>
<th>AC$^1$</th>
<th>RC$^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNL1</td>
<td>0.266</td>
<td>0.094</td>
</tr>
<tr>
<td>FNL2</td>
<td>0.297</td>
<td>0.125</td>
</tr>
<tr>
<td>FS0</td>
<td>0.266</td>
<td>0.094</td>
</tr>
</tbody>
</table>

$^1$AC: Absolute Coordinates  
$^2$RC: Relative Coordinates
Results: Horizontal Deflection in the 2D Case

First nonlinear formulation

![Graph showing horizontal deflection over time for ANCF, FNL1, and FNL2 methods. The graph includes a table with CPU times (s) for AC and RC methods.]

<table>
<thead>
<tr>
<th>Method</th>
<th>AC¹</th>
<th>RC²</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNL1</td>
<td>0.266</td>
<td>0.094</td>
</tr>
<tr>
<td>FNL2</td>
<td>0.297</td>
<td>0.125</td>
</tr>
<tr>
<td>FS0</td>
<td>0.266</td>
<td>0.094</td>
</tr>
</tbody>
</table>

¹AC: Absolute Coordinates
²RC: Relative Coordinates
Results: Horizontal Deflection in the 2D Case

**Foreshortening**

![Graph showing horizontal deflection over time](image)

**Table:** CPU–times (s)

<table>
<thead>
<tr>
<th>Method</th>
<th>AC(^1)</th>
<th>RC(^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNL1</td>
<td>0.266</td>
<td>0.094</td>
</tr>
<tr>
<td>FNL2</td>
<td>0.297</td>
<td>0.125</td>
</tr>
<tr>
<td>FS0</td>
<td>0.266</td>
<td>0.094</td>
</tr>
</tbody>
</table>

\(^1\) AC: Absolute Coordinates  
\(^2\) RC: Relative Coordinates
Results: Horizontal Deflection in the 2D Case

Foreshortening

<table>
<thead>
<tr>
<th>Method</th>
<th>AC(^1)</th>
<th>RC(^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FNL1</td>
<td>0.266</td>
<td>0.094</td>
</tr>
<tr>
<td>FNL2</td>
<td>0.297</td>
<td>0.125</td>
</tr>
<tr>
<td>FS0</td>
<td>0.266</td>
<td>0.094</td>
</tr>
</tbody>
</table>

\(^1\)AC: Absolute Coordinates
\(^2\)RC: Relative Coordinates
Results: Horizontal Deflection in the 3D Case

Substructuring

![Graph showing horizontal deflection over time with FEM and substructuring results for 3, 5, and 10 substructures.]
Results: Horizontal Deflection in the 3D Case

First nonlinear formulation
Results: Horizontal Deflection in the 3D Case

Foreshortening

![Graph showing Foreshortening](image-url)
Substructuring

Results: Vertical and Axial Displacements

- Local x coordinate (m)
- Local z coordinate (m)

- Time (s)

- FEM
- 10 sub.
- 5 sub.
- 3 sub.
Results: Vertical and Axial Displacements

First nonlinear formulation

![Graph showing vertical and axial displacements over time for FEM, FNL1, and FNL2 simulations.](image-url)
Results: Vertical and Axial Displacements

Foreshortening

![Graph showing foreshortening over time for different local coordinates (x, z)](image-url)

- Local x coordinate (m)
  - 9.92
  - 9.94
  - 9.96
  - 9.98
  - 10.00

- Local z coordinate (m)
  - -1.2
  - -1.0
  - -0.8
  - -0.6
  - -0.4

Time (s)

FEM
FS
### 3D spin–up beam results

<table>
<thead>
<tr>
<th>Method</th>
<th>SB10</th>
<th>SB20</th>
<th>FNL1</th>
<th>FNL2</th>
<th>FS0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU–time (s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC$^1$</td>
<td>–</td>
<td>–</td>
<td>0.271</td>
<td>0.286</td>
<td><strong>0.250</strong></td>
</tr>
<tr>
<td>RC$^2$</td>
<td>1.312</td>
<td>4.578</td>
<td>0.105</td>
<td>0.125</td>
<td><strong>0.105</strong></td>
</tr>
<tr>
<td>$Δx$ (mm)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.502</td>
<td><strong>0.320</strong></td>
<td>27.944</td>
<td>27.947</td>
<td>0.500</td>
<td></td>
</tr>
<tr>
<td>$Δy$ (mm)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.193</td>
<td><strong>1.481</strong></td>
<td>10.680</td>
<td>5.757</td>
<td>3.140</td>
<td></td>
</tr>
<tr>
<td>$Δz$ (mm)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.329</td>
<td><strong>0.742</strong></td>
<td>7.606</td>
<td>3.876</td>
<td>4.796</td>
<td></td>
</tr>
</tbody>
</table>

$^1$AC: Absolute Coordinates  $^2$RC: Relative Coordinates
Conclusions of the Fourth Chapter

- Linear FFR method cannot capture geometric stiffening effect

- Substructuring method
  - Best accuracy, increasing with the number of substructures
  - Easy implementation into FFR codes, no modifications required
  - High CPU–times if compared to other methods

- First nonlinear formulation
  - Very fast and easy to implement, only affects the $K$ matrix
  - No foreshortening $\implies$ highest error in axial direction
  - Axial modes are required

- Foreshortening
  - Almost as accurate and much faster than substructuring
  - No axial modes are required $\implies$ numerical integrator friendly
  - More involved implementation, requires preprocessing
Outline

Introduction

Formulation in Relative Coordinates

Inertia Shape Integrals

Geometric Stiffening

Conclusions and Future Research
Final Conclusions

- New semi-recursive $O(n^3)$ FFR formulation
  - Efficiency improvement for systems above 25 variables
  - More robust than the formulation in absolute coordinates
  - More involved implementation

- Shape integrals preprocessing implemented in both formulations
  - Always more efficient than $B^*$ matrix projection
  - Projection is much simpler and fast enough for small models
  - It is also more convenient for including foreshortening

- Three methods for modeling nonlinear beams have been tested
  - Substructuring is the most accurate approach
  - The $K_G$ method is extremely simple and yields acceptable results
  - Foreshortening obtains the best efficiency/accuracy ratio
Future Research

- Study of different model reduction methods
  - Krylov subspaces are based on response characteristics

- Mode selection techniques
  - The selection of mode shapes is left to the analyst
  - Development of automated techniques

- Further optimization of the calculation of the inertia terms
  - Check relative weights of the different terms
  - The number of operations depends on the reference conditions