A Collaborative Environment for Flexible Development of MBS Software

Manuel J. González Castro

A thesis submitted for the degree of Doctor Ingeniero Industrial

University of A Coruña

Ferrol, April 2005
Outline

1. Introduction
2. Data modeling
3. Benchmarking
4. Simulation software
5. Conclusions
Introduction

- MBS dynamics is an active research subject:
  - Many journal papers per year
  - Increasing number of conferences

- Many researchers working on open fields

- Development of new simulation methods:
  - Increase performance for real time
  - Handle complex non-linear aspects (contact-impact, friction, ...)

ECCOMAS Thematic Conference
Multibody Dynamics 2005
Madrid, June 2005

Fifth ASME International Conference on Multibody Systems, Nonlinear Dynamics and Control
Long Beach, September 2005
A Collaborative Environment for Flexible Development of MBS Software
Manuel J. González Castro
A thesis submitted for the degree of Doctor Ingeniero Industrial
April 20th, 2005 - Ferrol, Spain
Motivation

• Needed by scientific research:
  – Avoid duplication of efforts
  – Streamline research

• Needed by industry:
  – Products are very complex
  – Several teams must work together

• Needed by governments:
  – Requisite for some funding instruments

Research MBS needs tools that support collaboration
Introduction

Scope and objectives

Real system

New method

MBS model

Software

Results

Develop tools and standards to support collaboration

Model

Implement

Measure efficiency & accuracy

performance
1. Introduction
2. Data models
3. Benchmarking
4. Simulation software
5. Conclusions
Introduction

- **Neutral** data models are essential to exchange engineering product data
  - Avoid interoperability costs

- **There is no neutral data format for MBS**
  - Few users, one market leader
  - Interoperability costs are low

- The situation is changing
  - MBS community must address the problem as soon as possible

---

Data models

---

Evolution of CAD and CAE market shares

- **CAD**
- **CAE**

10% 20% 30% 40% 50% 60%


Evolution of CAD and CAE market shares
State of the art

- Engineering Product data
  - **STEP** (ISO 10303) is the current standard
  - Solves the exchange of CAD data
  - Currently being extended to CAE data: FEA, CFD, electronics, ...

- Multibody systems
  - German standardization efforts in the 1990s: DAMOS-C, MechaSTEP
  - Commercial software uses **proprietary data formats**

- **XML** (eXtensible Markup Language)
  - Emerging technology, very successful in other fields
  - Very easy to use
### Data models

**Evaluation of commercial software**

<table>
<thead>
<tr>
<th>Feature \ Preprocessor</th>
<th>ADAMS v.2003</th>
<th>SYMPACK v.8.6</th>
<th>DADS v.9.6</th>
<th>RecurDyn v.5.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model database format</td>
<td>Primary: Binary file</td>
<td>Text files</td>
<td>Binary file</td>
<td>Binary file</td>
</tr>
<tr>
<td></td>
<td>Secondary: Text file (.adm or .cmd)</td>
<td>-</td>
<td>Text file</td>
<td>-</td>
</tr>
<tr>
<td>Imports MBS models in other formats</td>
<td>No</td>
<td>No</td>
<td>ADAMS (.adm)</td>
<td>ADAMS (.adm and .cmd)</td>
</tr>
<tr>
<td>Exports MBS models in other formats</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Formalism-independent modeling</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Sub-models</td>
<td>No</td>
<td>Yes (only 1 level)</td>
<td>No</td>
<td>Yes (needs special preprocessing)</td>
</tr>
<tr>
<td>Units systems</td>
<td>MLT based</td>
<td>Not MLT based</td>
<td>MLT based</td>
<td>MLT based</td>
</tr>
<tr>
<td>Units scope</td>
<td>Global</td>
<td>Global</td>
<td>Global</td>
<td>Global</td>
</tr>
<tr>
<td>Parametric models</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Evaluation of STEP and XML

<table>
<thead>
<tr>
<th>Feature</th>
<th>STEP</th>
<th>XML</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Design</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Support for object-oriented modeling</td>
<td>●●●</td>
<td>●</td>
</tr>
<tr>
<td>Support for rule and constraint definition</td>
<td>●●●</td>
<td>●●</td>
</tr>
<tr>
<td>Support for hierarchical structures in the data model</td>
<td>●</td>
<td>●●●</td>
</tr>
<tr>
<td>Modeling language easy to learn</td>
<td>●●</td>
<td>●●●</td>
</tr>
<tr>
<td>Availability of data models for product data</td>
<td>●●●</td>
<td>●</td>
</tr>
<tr>
<td>Availability of data models for MBS</td>
<td>●●</td>
<td>●</td>
</tr>
<tr>
<td>Modular and configurable data models</td>
<td>●</td>
<td>●●</td>
</tr>
<tr>
<td><strong>Implementation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code generation</td>
<td>●●●</td>
<td>●</td>
</tr>
<tr>
<td>Human-readable physical file format</td>
<td>●●</td>
<td>●●●</td>
</tr>
<tr>
<td>Support for hierarchical structures in the file format</td>
<td>●</td>
<td>●●●</td>
</tr>
<tr>
<td>Quantity and quality of available supporting tools</td>
<td>●</td>
<td>●●●</td>
</tr>
<tr>
<td>Cost of available supporting tools</td>
<td>●</td>
<td>●●●</td>
</tr>
<tr>
<td>Quantity and quality of available documentation</td>
<td>●●</td>
<td>●●●</td>
</tr>
<tr>
<td>Cost of available supporting documentation</td>
<td>●</td>
<td>●●●</td>
</tr>
</tbody>
</table>

- good
- medium
- good

A Collaborative Environment for Flexible Development of MBS Software
Manuel J. González Castro

A thesis submitted for the degree of Doctor Ingeniero Industrial
April 20th, 2005 - Ferrol, Spain
Data models

- Information is decoupled to facilitate data reuse:
  - Model
  - Analysis
  - Method

- Support for:
  - Sub-models
  - Units of measure
  - Parametric models

- Modeling language with a modular design
  - Easily extensible and configurable
Data models

Application Program Interface (API)

- C++ programming library to read/write XML data files
  - Simplifies file processing
  - Object-oriented

```c++
// create empty simulation job
Job job;

// read XML file
XmlReader reader("doc.xml");
reader.read(job);

// examine job content
Model* m = job->getModel();
// ...
```

C++ for reading an XML simulation job
A Collaborative Environment for Flexible Development of MBS Software
Manuel J. González Castro

A thesis submitted for the degree of Doctor Ingeniero Industrial
April 20th, 2005 - Ferrol, Spain

Support for HTTP and FTP

Data models

Models
- model-1.xml
- model-2.xml
- model-3.xml

Analyses
- analysis-1.xml
- analysis-2.xml
- analysis-3.xml

Methods
- method-1.xml
- method-2.xml
- method-3.xml

Driver XML document

<job>
  <xi:include href="model-3.xml"/>
  <task>
    <xi:include href="analysis-1.xml"/>
    <xi:include href="method-2.xml"/>
  </task>
</job>

MBS simulation software

Including system
Data models

Automatic generation of XML files

- Plug-in for I-DEAS (CAD/CAE/CAM system)
- The MBS is modeled in the pre-processor
- The corresponding XML model is exported
- Due to I-DEAS limitations, joints and forces cannot be exported
  - Not useful
  - Serves as proof of concept of the idea
Conclusions

- Evaluation of commercial MBS software
  - Poor interoperability
  - Commercial data formats do not support collaboration

- Evaluation of STEP and XML as neutral data formats for multibody systems
  - STEP has better capabilities for design
  - XML seems to be much more easier to implement

- Prototype implementation of an XML-based data format for MBS
  - Simple yet powerful
  - Excellent capabilities for data exchange and reuse
  - XML proved to be a powerful, cheap and easy-to-use technology
Data models

- STEP still has some important advantages
  - Large library of models (CAD, FEA, ...)
  - Data models are more robust

An industrial-strength data model for MBS must use both STEP and XML

- Some international efforts to merge STEP and XML are under progress
  - Apply them to MBS
  - Very interesting and promising field
  - Needs cooperation at international level
A Collaborative Environment for Flexible Development of MBS Software
Manuel J. González Castro

Benchmarking

1. Introduction
2. Data models
3. Benchmarking
4. Simulation software
5. Conclusions

Benchmarking performance

Model

Real system

New method

MBS model

Implement

Software

Results

Measure efficiency & accuracy
Motivation

- MBS researchers:
  
  "I have developed a new simulation method. How good is it compared with others?"

- MBS users:
  
  "I need to simulate this system. Which method should I use?"
Benchmarking

- No easy answers
  - Efficiency depends on many inter-related factors

- Researchers report performance using different:
  - Models and analysis conditions
  - Accuracy in the solution

- Results are scattered and difficult to collect

<table>
<thead>
<tr>
<th>Time-step</th>
<th>CPU time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>20</td>
</tr>
<tr>
<td>0.001</td>
<td>45</td>
</tr>
</tbody>
</table>
Benchmarking

Objectives

- System to measure performance:
  - Standard problem collection
  - Reference solutions
  - Clear procedure to measure efficiency

- System to share performance measures
  - Collect, organize and share information
  - Centralized
  - Public
  - WWW seems very appropriate
Benchmarking

Problem collection

- Each problem describes the model, the analysis and measured coordinates

- Divided in categories
  - “Basic problems”
    - Small, isolate a particular characteristic
    - Need little time investment (important for a standard benchmark)
  - “Industrial applications”
    - Complex, real-life problems
    - Involving several complex phenomena together
    - Demonstrations for industry
### Benchmarking

#### Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>Double pendulum (2D)</td>
<td>Example, didactic problem</td>
</tr>
<tr>
<td>A01</td>
<td>Double pendulum (3D)</td>
<td>High accelerations</td>
</tr>
<tr>
<td>A02</td>
<td>Double four bar mechanism</td>
<td>Singular positions</td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td>Very small time scale</td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

#### Basic problems for rigid MBS
## Benchmarking

### Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td><strong>Double pendulum (2D)</strong></td>
<td><strong>Example, didactic problem</strong></td>
</tr>
<tr>
<td>A01</td>
<td>Double pendulum (3D)</td>
<td>High accelerations</td>
</tr>
<tr>
<td>A02</td>
<td>Double four bar mechanism</td>
<td>Singular positions</td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td>Very small time scale</td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

- **2 d.o.f.**
- Only gravity effects
- Duration: 15 s
- Measure end point coordinates
- Example problem
# Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>Double pendulum (2D)</td>
<td>Example, didactic problem</td>
</tr>
<tr>
<td>A01</td>
<td><strong>Double pendulum (3D)</strong></td>
<td><strong>High accelerations</strong></td>
</tr>
<tr>
<td>A02</td>
<td>Double four bar mechanism</td>
<td>Singular positions</td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td>Very small time scale</td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

**6 d.o.f.**

- Gravity effects
- Duration: 15 s
- Measure end point coordinates
- High accelerations (chaotic movement)
- Needs very accurate methods
## Benchmarking

### Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>Double pendulum (2D)</td>
<td>Example, didactic problem</td>
</tr>
<tr>
<td>A01</td>
<td>Double pendulum (3D)</td>
<td>High accelerations</td>
</tr>
<tr>
<td><strong>A02</strong></td>
<td><strong>Double four bar mechanism</strong></td>
<td><strong>Singular positions</strong></td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td>Very small time scale</td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

**1 d.o.f.**

Gravity effects

Duration: 15 s

Measure coordinates of p1

Singular configuration at horizontal position: 3 d.o.f.

*Bayo and Avello, 1994*
### Benchmarking

#### Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>Double pendulum (2D)</td>
<td>Example, didactic problem</td>
</tr>
<tr>
<td>A01</td>
<td>Double pendulum (3D)</td>
<td>High accelerations</td>
</tr>
<tr>
<td>A02</td>
<td>Double four bar mechanism</td>
<td>Singular positions</td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td><strong>Very small time scale</strong></td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

1 d.o.f.

Applied torque

Duration: 0.15 s

Measure coordinates of p4

**Very small time scale**

*Schiehlen, 1990*
### Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>Double pendulum (2D)</td>
<td>Example, didactic problem</td>
</tr>
<tr>
<td>A01</td>
<td>Double pendulum (3D)</td>
<td>High accelerations</td>
</tr>
<tr>
<td>A02</td>
<td>Double four bar mechanism</td>
<td>Singular positions</td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td>Very small time scale</td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

#### Benchmarking

- **1 d.o.f.**
- Gravity effects
- Duration: 10 s
- Measure coordinates of p3
- Redundant equations (Grübler: 0 d.o.f.)

*García de Jalón & Bayo, 1994*
### Problems in group A

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>Double pendulum (2D)</td>
<td>Example, didactic problem</td>
</tr>
<tr>
<td>A01</td>
<td>Double pendulum (3D)</td>
<td>High accelerations</td>
</tr>
<tr>
<td>A02</td>
<td>Double four bar mechanism</td>
<td>Singular positions</td>
</tr>
<tr>
<td>A03</td>
<td>Andrew’s mechanism</td>
<td>Very small time scale</td>
</tr>
<tr>
<td>A04</td>
<td>Bricard’s mechanism</td>
<td>Redundant equations</td>
</tr>
<tr>
<td>A05</td>
<td>Bicycle with rear suspension</td>
<td>Stiff system</td>
</tr>
</tbody>
</table>

- **1 d.o.f.**
- **Applied torque**
- **Duration:** 30 s
- **Measure coordinates of p1**
- **Stiff suspension spring**

*Good & McPhee, 1999*
Benchmarking

Problems in group B

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>B01</td>
<td>Iltis vehicle</td>
<td>Automotive</td>
</tr>
<tr>
<td>B02</td>
<td>Dornier’s antenna</td>
<td>Aerospace</td>
</tr>
<tr>
<td>B03</td>
<td>Human body</td>
<td>Biomechanics</td>
</tr>
<tr>
<td>B04</td>
<td>Puma robot</td>
<td>Robotics (serial)</td>
</tr>
<tr>
<td>B05</td>
<td>Stewart platform</td>
<td>Robotics (parallel)</td>
</tr>
</tbody>
</table>

Industrial applications for rigid MBS
Benchmarks

Reference solutions – Group A

- Solved with ADAMS/Solver, some of them also with Matlab
- Different methods were used to ensure convergence to the right solution
- The reference solution includes all the time-history of the measured coordinates
Benchmarking

- Reference solutions difficult to find

- Example: Iltis benchmark
  - ADAMS solutions vs. published solutions (plots)
  - “Good” agreement, but...
  - Which one is the reference solution? The average?

- To be fair, more solvers should be used:
  - Simpack, Recurdyn, ...
Benchmarking

Solve the problem as fast as possible within the required accuracy

- Accuracy is measured with L2-norm:

\[ e_j(t_i) = \left| \frac{y_j(t_i) - y_{j_{ref}}(t_i)}{y_{j_{ref}}(t_i)} \right| \]

\[ e_{2,2} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \frac{1}{n} \sum_{j=1}^{n} (e_j(t_i))^2} \]

- Reasonable error levels were determined from work-precision plots

\( e = 0.1\% \) (high)

\( e = 10\% \) (low)
Benchmarking

How to measure performance

- Efficiency of a simulation is computed with the Software Performance Ratio (S.P.R.):

\[
S.P.R._{\text{test problem } i} = \left(\frac{1}{H.P.R.}\right) \cdot \frac{\text{simulation time}_{\text{test problem } i}}{\text{CPU-time}_{\text{test problem } i}}
\]

- Tries to remove dependency from:
  - Simulation duration
  - Computer
Benchmarking

- Documentation
  - Specifications (HTML, PDF)
  - Reference solutions (numeric, plot, movie)
- Results submission
  - Only registered users (login required)
  - Detailed information about the simulator
  - Users can delete their results
- Results querying
  - Criteria and filters
  - HTML reports with graphic

http://lim.ii.udc.es/mbsbenchmark
BENCHMARKING

APPLICATION TO ADAMS

- The benchmark has been applied to ADAMS/Solver

- Numerical experiments with different:
  - 11 simulation methods
  - 4 solver versions (release, programming language)
  - 2 computers (single-processors, dual-processor)

- Results:
  - Problem A05 is too easy
  - The rest of the problems are good benchmarks
  - The precision level is important
Benchmarking

• Benchmark for MBS dynamics
  – Fully documented problems for rigid MBS
  – Simple procedure to measure efficiency

• Web-based system to manage performance data
  – Very useful to analyze information
  – Public, centralized, easy to use

• Application to a commercial software (ADAMS)
  – Validation of the proposed benchmark
  – Base-line results for future comparisons with other solvers
Benchmarks

Future work

- Extend the problem collection
  - Find reference solutions for “Industrial applications”
  - Include other phenomena: flexibility, contact-impact, ...

- Automate the benchmarking procedure
  - Useful to control quality of software releases

- Apply the benchmark to other simulation codes
  - Commercial
  - Academic
Simulation software

1. Introduction
2. Data models
3. Benchmarking
4. Simulation software
5. Conclusions
Simulation software

Motivation

• Usually, engineers do not use software design techniques
  – Code developed ad-hoc to solve a particular problem
  – Bad programming style, code difficult to reuse

• General MBS simulation software can become very complex
  – Needs methods for software engineering
  – Needs collaboration between programmers
Simulation software

Objective

- Design a general-purpose, generic MBS simulation software
  - Not tied to a particular formulation
  - Support for multiple simulation methods
  - Modular and extensible

- Select the right tools and techniques
  - Development environment
  - Programming language, numerical libraries, ...

- Deploy the system and train colleagues
Simulation software

Evaluation of CASE tools

• CASE = Computer Aided Software Engineering
  – Code is generated from graphical models

• Evaluation of a commercial tool:
  – Very difficult, needs too customization
  – Not adequate for academic environments

• An open source project host is best suited:
  – Source control
  – Bug tracking, task management, ...
  – After evaluation, Berlios was selected
<table>
<thead>
<tr>
<th><strong>Simulation software</strong></th>
<th><strong>Development environment</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>License</td>
<td>Open Source (GPL)</td>
</tr>
<tr>
<td>Programming language</td>
<td>C++ / Fortran</td>
</tr>
<tr>
<td>Compiler</td>
<td>Visual Studio 7/ GNU GCC 3</td>
</tr>
<tr>
<td>Source control system</td>
<td>SNV</td>
</tr>
<tr>
<td>Documentation tool</td>
<td>Doxygen</td>
</tr>
<tr>
<td>Design tool</td>
<td>Poseidon UML</td>
</tr>
<tr>
<td>Management</td>
<td>XPlanner</td>
</tr>
<tr>
<td>Project Host</td>
<td>Berlios</td>
</tr>
<tr>
<td>XML parser</td>
<td>Libxml</td>
</tr>
<tr>
<td>Visualization System</td>
<td>OSG / OpenGL Performer</td>
</tr>
<tr>
<td>Math kernel (BLAS)</td>
<td>ATLAS</td>
</tr>
</tbody>
</table>
Simulation software

Design

A Collaborative Environment for Flexible Development of MBS Software
Manuel J. González Castro

A thesis submitted for the degree of Doctor Ingeniero Industrial
April 20th, 2005 - Ferrol, Spain
Simulation software

Current state

- Design and skeleton of the program (“base classes”) is finished
- General facilities
  - XML input/output and friendly error reports
- Formalisms
  - Support only global formulations based on natural coordinates
  - Library of joints
  - Automatic constraint generation for joints
- Numerical methods (solvers)
  - Matrix class library to wrap different:
    - Data structures: dense and sparse
    - Linear solvers (TAU, PHIPHACS, Harwell library, …)
- Generic interface for integrators
1. Introduction
2. Data modeling
3. Benchmarking
4. Simulation software
5. Conclusions
Conclusions

- The barriers to collaboration in MBS dynamics have been studied
- Extensive review of the state of the art
- Evaluation and selection of tools and technologies

- Solutions have been proposed to:
  - Neutral data format
  - Benchmarking system
  - Simulation software

- Prototypes have been proposed for all the systems
Conclusions

A unified simulation environment for real-time multibody systems dynamics with stress analysis and control (UDC)

Collaborative tools for multibody system dynamics (UDC-US, 10 researchers)

Future work

2000 2001 2002 2003 2004 2005 2006 2007


2001 - 2005