VEHICLE DYNAMIC MODELING AND SIMULATION: COMPARING A FINITE-ELEMENT SOLUTION TO A MULTI-BODY DYNAMIC SOLUTION

Paramsothy Jayakumar, PhD
Dynamics & Structures Modeling & Simulation
US Army RDECOM TARDEC
Warren, MI

Tamer Wasfy, PhD
Advanced Science and Automation Corp.
Indianapolis, IN
**Title:** VEHICLE DYNAMIC MODELING and SIMULATION: COMPARING A FINITE-ELEMENT TO A MULTI-BODY DYNAMIC SOLUTION

**Authors:** Paramsothy Jayakumar; Tamer Wasfy

**Performing Organization:**
Advanced Science and Automation Corp., 9714 Oakhaven Ct., Indianapolis, IN, 46256-8101

**Performing Organization Report Number:** #21096

**Sponsoring/Monitoring Agency:**
U.S. Army TARDEC, 6501 East Eleven Mile Rd, Warren, Mi, 48397-5000

**SPONSOR/MONITOR’S ACRONYM:** TARDEC

**Distribution/Availability Statement:** Approved for public release; distribution unlimited

**Confidentiality and Access:**
- Report: unclassified
- Abstract: unclassified
- This Page: unclassified

**Subject Terms:**
briefing charts

**Limitation of Abstract:** Public Release

**Number of Pages:** 23

---

For 2010 GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM (GVSETS) AUGUST 17-19
Motivation

- Army needs high-fidelity multibody dynamic modeling of wheeled and tracked vehicles for predicting:
  - Stability (going over a rough terrain at high-speed, lane change, etc.)
  - Mobility analysis (going over bumps and potholes).
  - Fatigue life and durability of various vehicle components.
Motivation (cont.)

- Two modeling and simulation approaches:
  - Traditional solution: Rigid/Flexible Multibody Dynamics (MBD) codes.
  - Alternate solution: Explicit Finite Element (FE) codes.

- Advantages of Explicit FE codes over MBD codes.
  - Multi-physics modeling such as fluid-structure interaction or thermal effects.
  - Same software environment to model flexible bodies and to predict stresses
    - However, accurate stress analysis requires refined mesh and takes significantly more CPU time.
  - Easily accommodate nonlinear material characteristics such as plasticity and fracture.
  - Can be easily parallelized.
  - Simulation time increases linearly with the DOFs.
Motivation (cont.)

• Disadvantages of explicit FE codes over MBD codes.
  – Most explicit FE codes use an incremental updated-Lagrangian solution formulation which can lead to solution drift.
  – Use inexpensive finite elements that use spurious modes control.
  – Calculated joint reaction forces exhibit high-frequency oscillations. MBD codes, on the other hand, produce accurate joint forces.
  – Integration time step ($\Delta t$) must be less than a critical time step for stability.
    • This requires a very small time step for stiff systems. MBD codes, on the other hand, can use implicit integrators that allow larger time steps.
Objective

• Benchmark the two modeling and simulation approaches.
  • Multibody dynamic code.
  • Explicit finite element code.

• Benchmark using two multibody dynamic systems.
  • A 7-link planar mechanism.
  • A spatial robotic manipulator.

• Comparisons are made of:
  • Body motion.
  • Joint constraint forces.
  • Conservation of energy.
  • CPU time.

• Conclusions are drawn regarding solution accuracy and efficiency of the two codes.
Explicit FE Code Formulation

• Equations of motion:

\[ M_K \ddot{x}_{Ki}^t = F_{s Ki}^t + F_{a Ki}^t \]

\[ I_{Kij} \ddot{\theta}_{Kj}^t = T_{s Ki}^t + T_{a Ki}^t - \left( \dot{\theta}_{Ki}^t \times (I_{Kij} \dot{\theta}_{Kj}^t) \right)_{Ki} \]

• Integrated using trapezoidal explicit integration formula:

\[ \dot{x}_{Kj}^t = \dot{x}_{Kj}^{t-\Delta t} + 0.5 \Delta t \left( \dot{x}_{Kj}^t + \dot{x}_{Kj}^{t-\Delta t} \right) \]

\[ x_{Kj}^t = x_{Kj}^{t-\Delta t} + 0.5 \Delta t \left( \dot{x}_{Kj}^t + \dot{x}_{Kj}^{t-\Delta t} \right) \]

\[ \dot{\theta}_{Kj}^t = \dot{\theta}_{Kj}^{t-\Delta t} + 0.5 \Delta t \left( \dot{\theta}_{Kj}^t + \dot{\theta}_{Kj}^{t-\Delta t} \right) \]

\[ \Delta \theta_{Kj}^t = 0.5 \Delta t \left( \dot{\theta}_{Kj}^t + \dot{\theta}_{Kj}^{t-\Delta t} \right) \]

\[ R_K^t = R_{K}^{t-\Delta t} \ R(\Delta \theta_{Ki}^t) \]
Explicit FE Code Formulation

- **Constraint modeling.**
  - Penalty formulation for modeling normal contact/joint forces.
    \[
    F_{c_i} = (k_p |\vec{d}| + c_p \dot{d}_i \dot{d}_i) \dot{d}_j
    \]
    \[
    d_i = x_{p1_i} - x_{p2_i}
    \]
    where \( k_p \): penalty stiffness, \( c_p \): penalty damping

- **Friction model.**
  - Asperity-spring friction model is used to model joint and contact friction.
Multibody Dynamic Code Formulation

- Equations of motion:
  \[
  \begin{bmatrix}
  M & \frac{\partial \Phi^T}{\partial q} \\
  \frac{\partial \Phi}{\partial q} & 0
  \end{bmatrix}
  \begin{bmatrix}
  \dot{\mathbf{q}} \\
  \lambda
  \end{bmatrix} = \begin{bmatrix}
  \mathbf{Q} \\
  \gamma
  \end{bmatrix}
  \]

- Constraint equations:
  \[\Phi(q, t) = 0\]
  \[\frac{\partial \Phi}{\partial q} \dot{q} = \nu\]

- Forms a set of Differential-Algebraic Equations (DAEs).

- Solved using implicit integration methods such as the Backward Differentiation Formula.
  - Can take much larger time step than explicit methods.
  - Very advantageous for stiff systems.
  - Time step is only dictated by desired solution accuracy.
Benchmark Problem 1: 7-Link Planar Mechanism

- 2D mechanism with 7 rigid links.
- 1 DOF system.
- Driven by a motor torque applied at point $O$.
  - Torque is removed at time $= 0.1$ sec to assess the energy conservation.
- Total solution time $= 0.5$ sec.
Benchmark Problem 1:
7-Link Planar Mechanism

• Time step / CPU time comparison.

• Animation.

<table>
<thead>
<tr>
<th></th>
<th>MBD</th>
<th>FE</th>
<th>FE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time step (s)</td>
<td>1.0E-5</td>
<td>0.4E-5</td>
<td>0.1E-5</td>
</tr>
<tr>
<td>CPU time (s)</td>
<td>20.875</td>
<td>4.955</td>
<td>19.820</td>
</tr>
<tr>
<td>Position/velocity</td>
<td>1E-3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>error Tolerance</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• Link K1 rotation angle comparison.

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Beta (rad)</th>
<th>% Difference between Handbook &amp;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MBD</td>
</tr>
<tr>
<td>0.000E+00</td>
<td>-6.1994E-02</td>
<td>-6.2017E-02</td>
</tr>
<tr>
<td>5.000E-03</td>
<td>2.10881E-01</td>
<td>2.06715E-01</td>
</tr>
<tr>
<td>1.000E-02</td>
<td>2.16040E+00</td>
<td>2.14057E+00</td>
</tr>
<tr>
<td>1.500E-02</td>
<td>5.65556E+00</td>
<td>5.64598E+00</td>
</tr>
</tbody>
</table>
Benchmark Problem 1: 7-Link Planar Mechanism

- Link rotation angles comparison
Benchmark Problem 1: 7-Link Planar Mechanism

- Link velocities comparison.
 Benchmark Problem 1:  
7-Link Planar Mechanism

- Joint force comparison.

![Joint B X-Force on Link K3](image1)
![Joint B Y-Force on Link K3](image2)
- Effect of FE time step on joint forces.
Benchmark Problem 1: 7-Link Planar Mechanism

• FE joint force with ramped applied torque and spring pre-load.

**Graphs:**

Joint B X-Force on Link K3 (FE)

Joint B Y-Force on Link K3 (FE)

FE: \( \Delta t = 0.4E-5 \)
Benchmark Problem 1:
7-Link Planar Mechanism

- Energy comparison.

![Energy comparison plots](chart.png)
Benchmark Problem 2: 
Spatial Robotic Manipulator

- 3 rigid bodies connected using cylindrical and revolute joints.
- 5 DOF system.
- External loads: gravity and prescribed joint actuator forces/torques.
- End effector traces a straight line with a trapezoidal velocity profile.
- Total solution time = 2.0 sec.
Benchmark Problem 2: Spatial Robotic Manipulator

- Time step / CPU time comparison.

<table>
<thead>
<tr>
<th></th>
<th>MBD</th>
<th>FE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time step (s)</td>
<td>0.01</td>
<td>3.75E-5</td>
</tr>
<tr>
<td>CPU time (s)</td>
<td>0.172</td>
<td>1.29</td>
</tr>
</tbody>
</table>

- Animation.
Benchmark Problem 2: Spatial Robotic Manipulator

• Manipulator Position Comparison
Benchmark Problem 2: Spatial Robotic Manipulator

Manipulator Velocity Comparison
Benchmark Problem 2: Spatial Robotic Manipulator

• Manipulator Joint Force Comparison
Conclusions

• Two multibody dynamics benchmark problems were solved using
  - an explicit finite-element code and
  - an implicit multibody dynamics code.

• The two codes predict the same system motion.

• Joint reaction forces predicted by FE code have high-frequency oscillations due to the penalty method used.

• To eliminate high-frequency force oscillations when using FE code, applied forces/moments must be continuous and the simulation should start from static equilibrium.

• Implicit MBD codes are computationally more efficient than Explicit FE codes for stiff MBD systems.