A COMPARISON OF NUMERICAL METHODS FOR SIMULATING MULTIBODY SYSTEMS

By

Ehab Eshtaya

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
HONOURS IN MATHEMATICS
AT
DALHOUSIE UNIVERSITY
HALIFAX, NOVA SCOTIA
APRIL 2004

DALHOUSIE UNIVERSITY
FACULTY OF
MATHEMATICS AND STATISTICS

The undersigned hereby certify that they have read and recommend to the Faculty of science for acceptance a thesis entitled "**A Comparison of Numerical Methods for simulating Multibody Systems**" by **Ehab Eshtaya** in partial fulfillment of the requirements for the degree of **Honours in Mathematics**.

Dated: April 2004

Supervisor: ______________________
Dr. Raymond Spiteri

Readers: ______________________
Dr. Roman Smirnov

____

DALHOUSIE UNIVERSITY

Date: **April 2004**

Author: **Ehab Eshtaya**

Title: **A Comparison of Numerical Methods for simulating Multibody Systems**

Faculty: **Mathematics and Statistics**

Degree: **BSc.** Convocation: **April** Year: **2004**

Contents

List of Tables

List of Figures

Chapter 1

Examples of Multibody System Simulation

In this chapter we describe two multibody systems: a two-dimensional slider-crank and a two-link pendulum. For the two-link pendulum, the equations of motion are generated both in the form of ODEs as well as DAEs. We use MBSSIM to solve the DAEs, and *Matlab* and the Newmark method (**??**), (**??**) to solve ODEs. We begin with the two-dimensional slider crank.

1.1 Two-dimensional Slider Crank

The two-dimensional slider-crank is described in Example **??** of chapter 1. We choose the following parameters

$$\begin{aligned}
l_1 &= 1 \text{ m},\\
l_2 &= 3 \text{ m},\\
m_1 &= 1 \text{ kg},\\
m_2 &= 3 \text{ kg},\\
m_3 &= 0 \text{ kg},\\
g &= -9.86 \text{ ms}^{-2}.
\end{aligned}$$

The DynaFlex description of the slider-crank was provided in Chapter 2 (Figure **??**). We provide the equations of motion for the two-dimensional slider-crank in the form of (1.1)

$$\begin{aligned}
\mathbf{M}\ddot{\mathbf{q}} - \mathbf{\Phi}_{\mathbf{q}}^T\boldsymbol{\lambda} &= \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, t),\\
\mathbf{\Phi}(\mathbf{q}, t) &= 0.
\end{aligned} \tag{1.1}$$

In this example

$$
\begin{aligned}
\mathbf{M} &= \begin{pmatrix} \frac{1}{3}m_1 l_1^2 & 0 & 0 \\ 0 & \frac{1}{3}m_2 l_2^2 & -\frac{1}{2}m_2 l_2 \cos\beta \\ 0 & -\frac{1}{2}m_2 l_2 \cos\beta & m_3 + m_2 \end{pmatrix}, \\
\mathbf{q} &= \begin{pmatrix} \theta \\ \beta \\ s \end{pmatrix}, \\
\mathbf{\Phi_q} &= \begin{pmatrix} -l_1 \sin\theta & l_2 \cos\beta & -1 \\ l_1 \cos\theta & l_2 \sin\beta & 0 \end{pmatrix}, \\
\mathbf{F} &= \begin{pmatrix} -\frac{1}{2} g m_1 l_1 \cos\theta \\ \frac{1}{2} g m_2 l_2 \sin\beta \\ -\frac{1}{2} m_2 \dot{\beta}^2 l_2 \sin\beta \end{pmatrix}, \\
\mathbf{\Phi}(\mathbf{q}, t) &= \begin{pmatrix} l_1 \cos\theta + l_2 \sin\beta - s \\ l_1 \sin\theta - l_2 \cos\beta \end{pmatrix}.
\end{aligned}
$$

The initial conditions used for the two-dimensional slider-crank are given by

$$
\begin{aligned}
\theta_0 &= 1.570796327, \\
\beta_0 &= 1.230959417, \\
s_0 &= 2.828427124 \text{ m}, \\
\dot{\theta} &= 2.0 \text{ s}^{-1}, \\
\dot{\beta} &= 0 \text{ s}^{-1}, \\
\dot{s} &= 2.0 \text{ ms}^{-1}.
\end{aligned}
$$

The initial conditions must satisfy $\mathbf{\Phi}(\mathbf{q}, t) = 0$. For consistency, the initial velocities satisfy $\dot{\mathbf{\Phi}}(\mathbf{q}, \mathbf{t})$. A 10-second simulation of the slider crank is solved using MBSSIM. We provide numerical approximation for θ (Figure 1.1), β (Figure 1.2), and s (Figure 1.3).

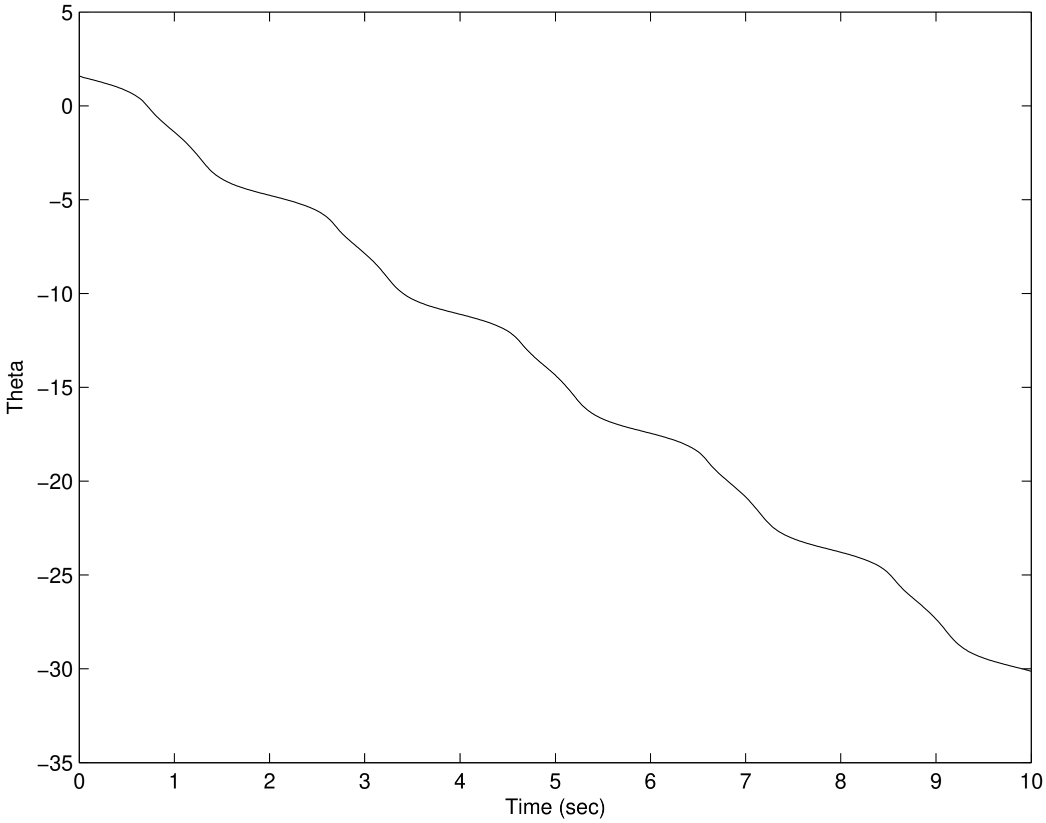

Figure 1.1: The 2D-Slider: plot of θ versus time

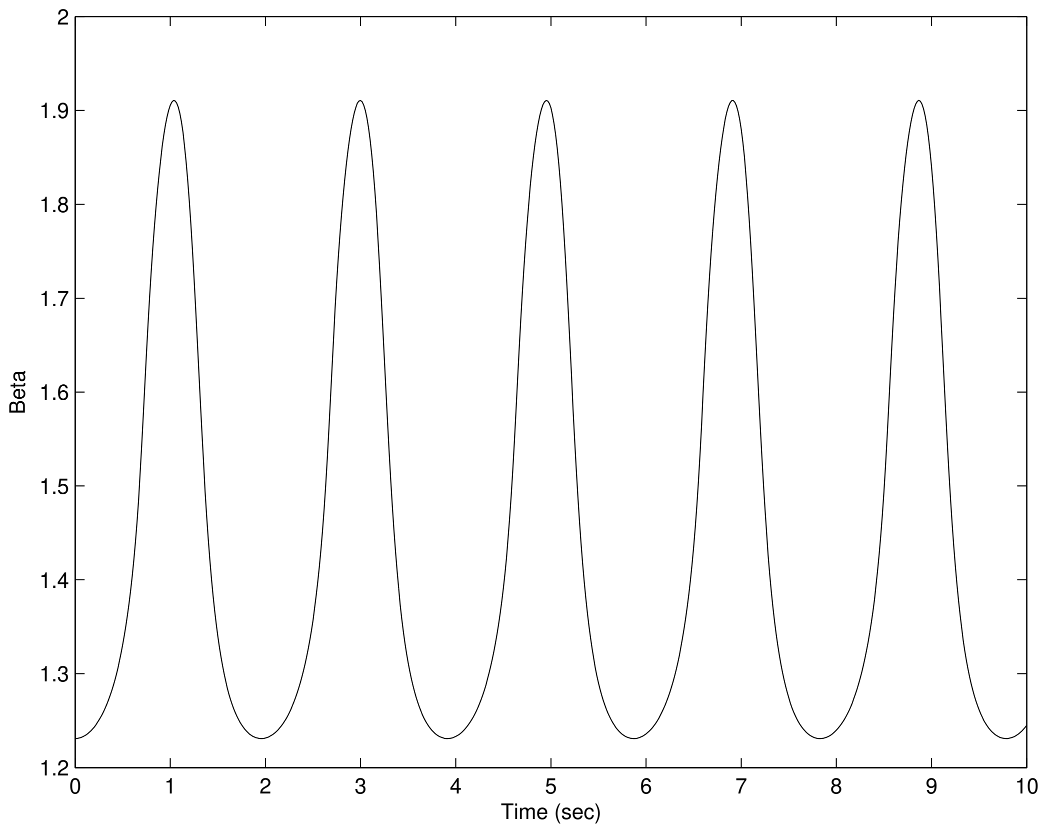

Figure 1.2: The 2D-Slider: plot of β versus time

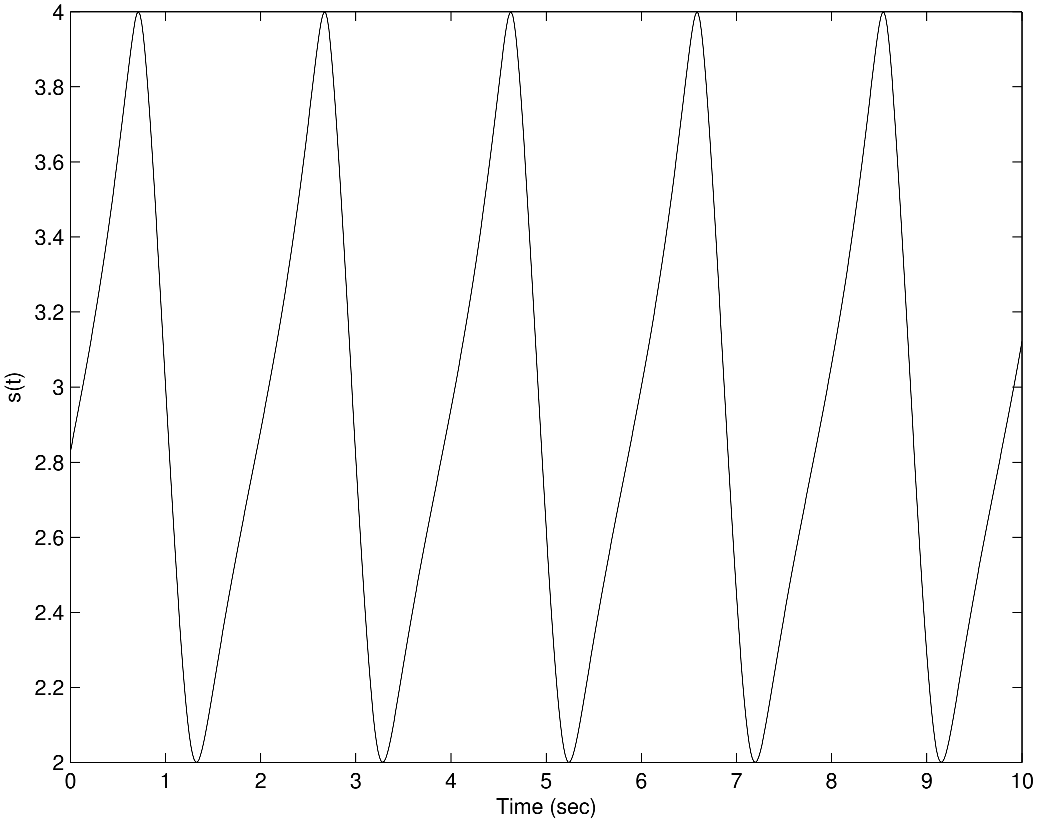

Figure 1.3: The 2D-Slider: plot of s versus time

The crank is given a small initial velocity to insure that the crank has negative rotation, i.e., clockwise direction. As illustrated in Figure 1.1, θ is changing in the negative direction, as expected. The value of β fluctuates between 1.23 and 1.91, corresponding to the positions where θ is at $\frac{\pi}{2} + 2\pi k$ and $-\frac{\pi}{2} + 2\pi k$, for $k \in \mathbb{Z}$. The distance between the origin and the slider s fluctuates between 2 and 4, corresponding to θ at $\pi + 2\pi k$ and $2\pi k$, for $k \in \mathbb{Z}$. The assigned tolerances for the solver are 10^{-3} for relative tolerance and 10^{-5} for absolute tolerance. Decreasing the tolerance makes little to no change in the final solution suggesting that the method is accurate. MBSSIM required 98 steps in total, 20 of which were rejected and 78 were accepted for the tolerance mentioned.

1.2 The two-link pendulum

In this section we describe the two-link pendulum (Figure 1.4). The two bodies are assumed to be rigid with masses m_1, m_2, and lengths l_1, l_2 respectively. The two joints about θ and β are assumed to be revolute, i.e., one degree of freedom. Because it is possible to describe the two-link pendulum using an open-loop system, we model the two-link pendulum using two separate models. The first is by generating a set of ODEs using joint coordinates, i.e., θ and β, and the second is by generating a set of DAEs using Cartesian coordinates.

1.2.1 ODE model for two-link pendulum

As discussed in Chapter 2, in order to generate equations of motion in the form of ODEs, we require that the description of the multibody system to be given by an open-loop system. The construction of the DynaGUI description of the two-link manipulator is similar to that of the two-dimensional slider crank provided in Chapter 2; hence, we only provide the final DynaGUI description (Figure 1.5).

The model is then read by *Maple* and the equations of motion are generated using DynaFlex. We present the equations of motion in the following form

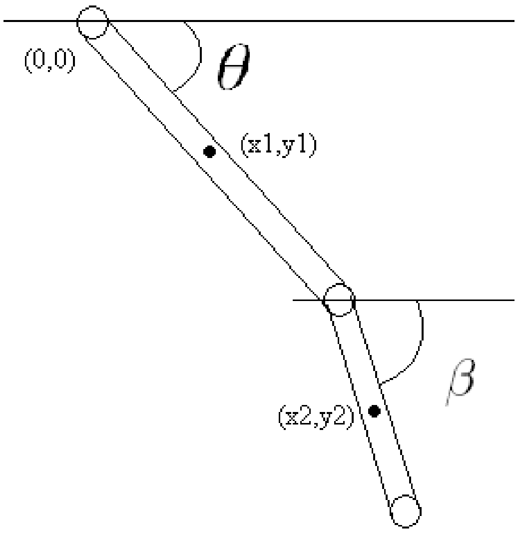

Figure 1.4: The Two-Link Pendulum

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{F} \tag{1.2}$$

where

$$\begin{aligned}
\mathbf{M} &= \begin{pmatrix} \frac{1}{3}m_2 l_2^2 + m_2 l_1 l_2 \cos\beta + \frac{1}{3}m_1 l_1^2 + m_2 l_1^2 & \frac{1}{2}m_2 l_1 l_2 \cos\beta + \frac{1}{3}m_2 l_2^2 \\ \frac{1}{2}m_2 l_1 l_2 \cos\beta + \frac{1}{3}m_2 l_2^2 & \frac{1}{3}m_2 l_2^2 \end{pmatrix} \\
\ddot{\mathbf{q}} &= \begin{pmatrix} \ddot{\theta} \\ \ddot{\beta} \end{pmatrix} \\
\mathbf{F} &= \begin{pmatrix} \begin{pmatrix} m_2 \dot{\beta}\dot{\theta} l_1 l_2 \sin\beta + \frac{1}{2}m_2 \dot{\beta}^2 l_1 l_2 \sin\beta - \frac{1}{2} g m_1 l_1 \cos\theta \\ -\frac{1}{2}m_2 l_2 g \cos\theta\cos\beta + \frac{1}{2}m_2 l_2 g \sin\theta\sin\beta - g m_2 l_1 \cos\theta \end{pmatrix} \\ \frac{1}{2}m_2 l_2 \left(-\dot{\theta}^2 l_1 \sin\beta - g\cos\theta\cos\beta + g\sin\theta\sin\beta\right) \end{pmatrix}
\end{aligned}$$

The multibody system is given the following parameters and initial conditions

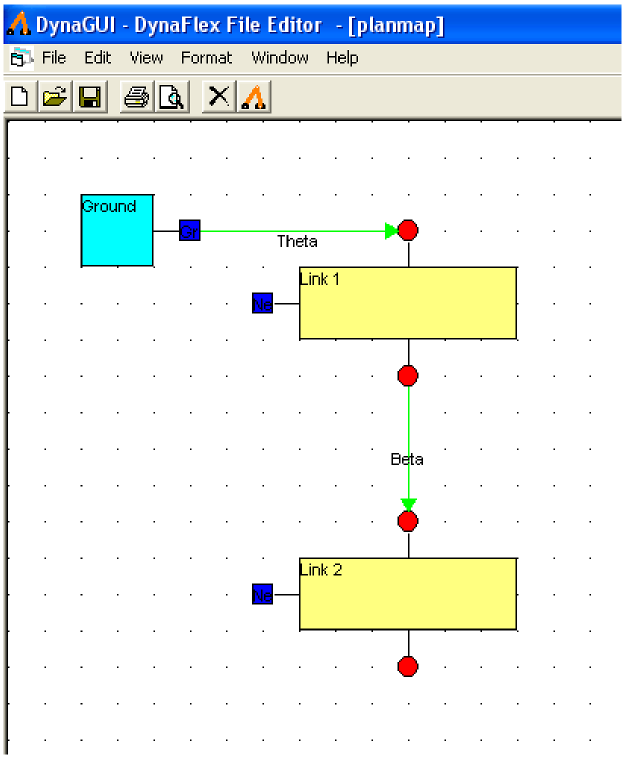

Figure 1.5: Description of the two-link manipulator using DynaGUI for the ODE model

$$\begin{aligned}
m_1 &= 3 \text{ kg}, \\
m_2 &= 1 \text{ kg}, \\
l_1 &= 3 \text{ m}, \\
l_2 &= 1 \text{ m}, \\
g &= -9.86 \text{ ms}^{-2}, \\
\theta_0 &= 0, \\
\beta_0 &= 0, \\
\dot{\theta}_0 &= 0 \text{ s}^{-1}, \\
\dot{\beta}_0 &= 0 \text{ s}^{-1}.
\end{aligned}$$

The ODE is then solved using *Matlab*'s `ode45` and the average acceleration Newmark method (**??**) from $t_0 = 0$ to $t_f = 10$. `ode45` had 408 successful steps and 55 failed attempts while solving the ODE with relative tolerance of 10^{-3} and absolute tolerance of 10^{-5}. The numerical approximation of the ODEs using the Newmark method with 408 steps ($Nsteps$) has a similar pattern to that approximated by `ode45`, but they are clearly different. This can be seen from the plot of θ versus time in Figure 1.6. The two plots are extremely close in the first 3 seconds. However, a phase shift begins to occur after the 3 second mark; furthermore, their amplitudes are slightly different.

The difference between the numerical approximation of the two methods is more evident when considering the plot of β versus time (Figure 1.7). Once again the two methods produce very close results for the first 3 seconds, but deviates slowly from each for $t > 3$. We double the number of steps ($2Nsteps$) for the Newmark method and plot the approximation (Figure 1.7). As illustrated in Figure 1.7, the numerical approximation of the Newmark method ($2N$) is a closer approximation to the solution produced by `ode45`. Because decreasing the step size of the Newmark method brings us closer to the `ode45` solution, we assume that `ode45` provides a more accurate result with smaller number of steps.

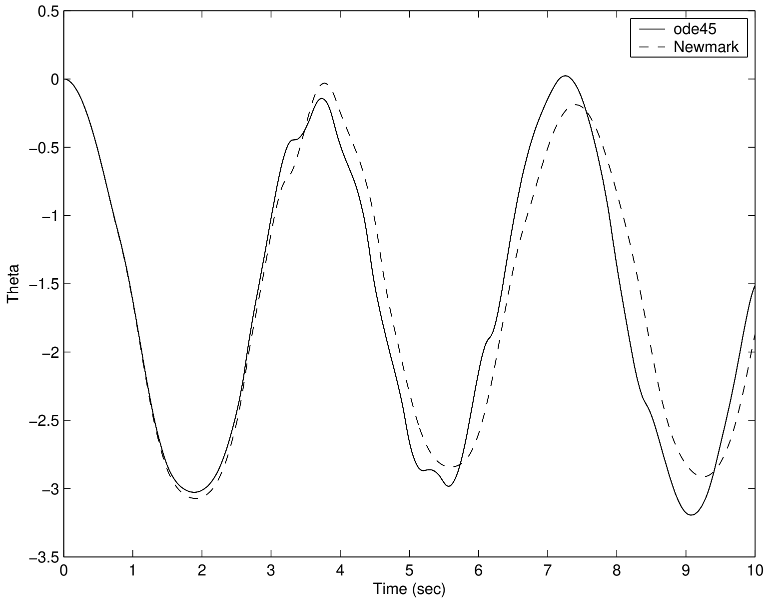

Figure 1.6: Plot of θ versus time for the two-link pendulum

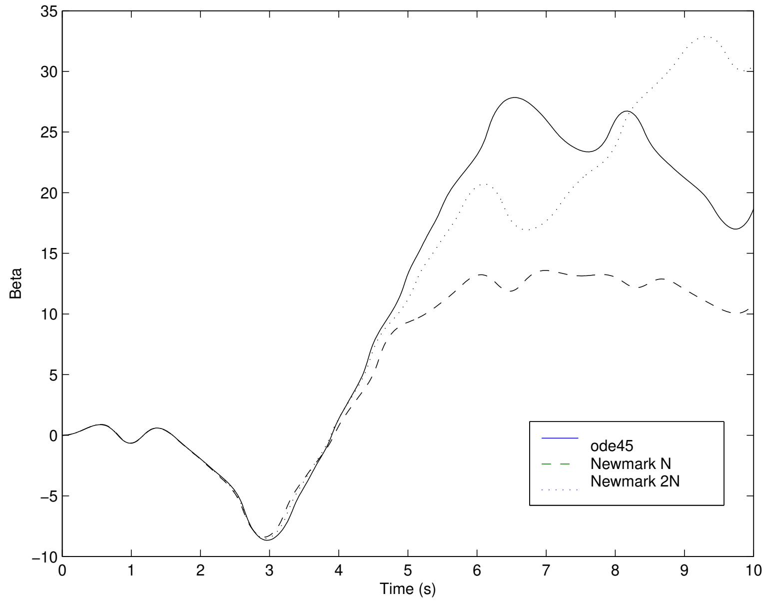

Figure 1.7: Plot of β versus time for the two-link pendulum

It should be noted that in practice most multibody systems do not produce an open-loop system. We now consider the modelling of the two-link pendulum by producing DAEs to describe the equations of motion by DynaFlex.

1.2.2 DAE model of the two-link pendulum

As discussed in Chapter 2 we require a closed-loop in the description of the DynaGUI graph to get a set of DAEs for the equations of motion. This adjustment is made from the ODE model by creating two planar joints each starting at the origin of the pendulum and terminating at the center of mass of $Link_1$ and $Link_2$ respectively. A planar joint restricts the motion of one body to be in the plane of another body's reference frame. Because this is a two-dimensional model, no additional physical constraints are constructed. The two new planar joint are placed in the tree of the DynaFlex formulation, and the original revolute joints are placed in the cotree (Figure 1.8).

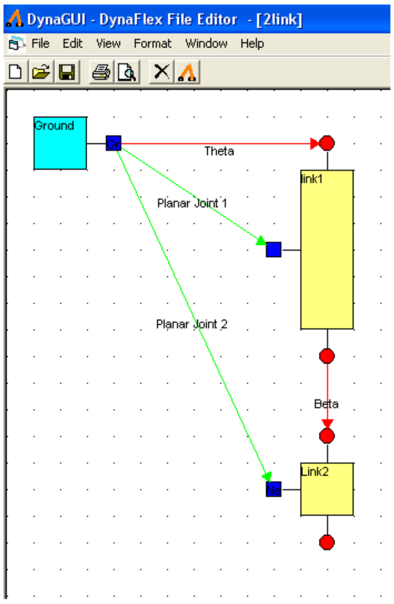

Figure 1.8: Description of the two-link manipulator using DynaGUI for the DAE model

A planar joint uses three generalized coordinates; $x(t)$ describing the x-coordinate of the center of mass of the link relative to the origin (Datum); $y(t)$ describes the y-coordinate of the center of mass of the link relative to the origin; $\phi(t)$ describes the angles illustrated in Figure 1.4 depending on which link is chosen. Hence for the DAE model of the two-link pendulum we have six generalized coordinates, three for each planar joint. Because this multibody system has two degrees of freedom we require four constraint equations. The equations of motion for the pendulum are provided in the form of (**??**), where

$$\mathbf{M} = \begin{pmatrix} m_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & m_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{m_1\, l_1{}^2}{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & m_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{m_2\, l_2{}^2}{12} \end{pmatrix}$$

$$\ddot{\mathbf{q}} = \begin{pmatrix} \ddot{x}_1 \\ \ddot{y}_1 \\ \ddot{\theta} \\ \ddot{x}_2 \\ \ddot{y}_2 \\ \ddot{\beta} \end{pmatrix}, \quad \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, t) = \begin{pmatrix} 0 \\ -g\, m_1 \\ 0 \\ 0 \\ -g\, m_2 \\ 0 \end{pmatrix}$$

$$\mathbf{\Phi}(\mathbf{q}, t) = \begin{pmatrix} 1/2\, \cos\theta l_1 - x_1 \\ 1/2\, \sin\theta l_1 - y_1 \\ 1/2\, \cos\theta l_1 + 1/2\, \cos\beta l_2 + x_1 - x_2 \\ 1/2\, \sin\theta l_1 + 1/2\, \sin\beta l_2 + y_1 - y_2 \end{pmatrix}$$

This system of DAEs is approximated by MBSSIM. For comparison, the ODE model approximated with `ode45` is plotted with the MBSSIM approximation for θ and β (Figures 1.9-1.10); both solvers used relative tolerance of 10^{-3} and absolute tolerance of 10^{-5}. The plot of θ versus time is consistent between the two models with little variation. The plot of β versus time (Figure 1.10) is clearly different between the two methods of simulation. A second plot was made, where the relative and absolute tolerance were changed to 10^{-5} and 10^{-6} respectively. Interestingly, the solution for the DAE model stayed more or less the same while the ODE model solved by `ode45` displayed a great change (Figure 1.11). The consistency of the solution given by MBSSIM suggests that the DAE model is more reliable. However, it is difficult to make any conclusion about the differences between the two plots, because there are two variables to consider. First, one model is based on index-3 DAEs whereas the other is based on a system of ODEs. Second, each formulation is solved using two independent methods.

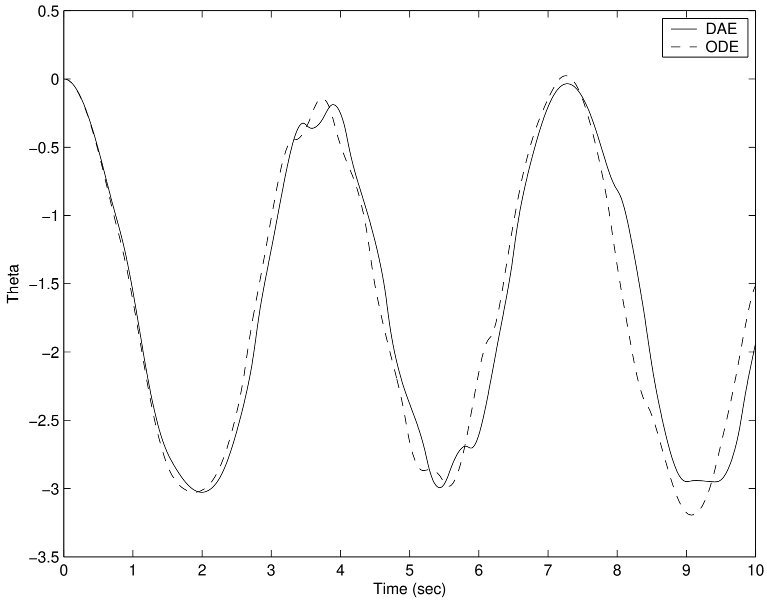

Figure 1.9: Plot of θ versus time for the DAE model of the two-link pendulum

In conclusion we notice that `ode45` is more accurate than the Newmark method in

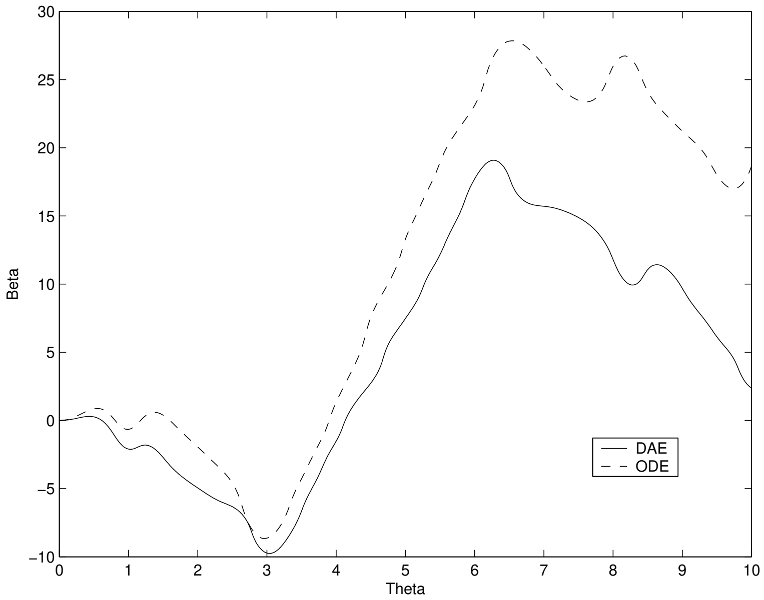

Figure 1.10: Plot of β versus time for the DAE model of the two-link pendulum with Rel. Tol. 10^{-3}, and Abs. Tol. 10^{-5}

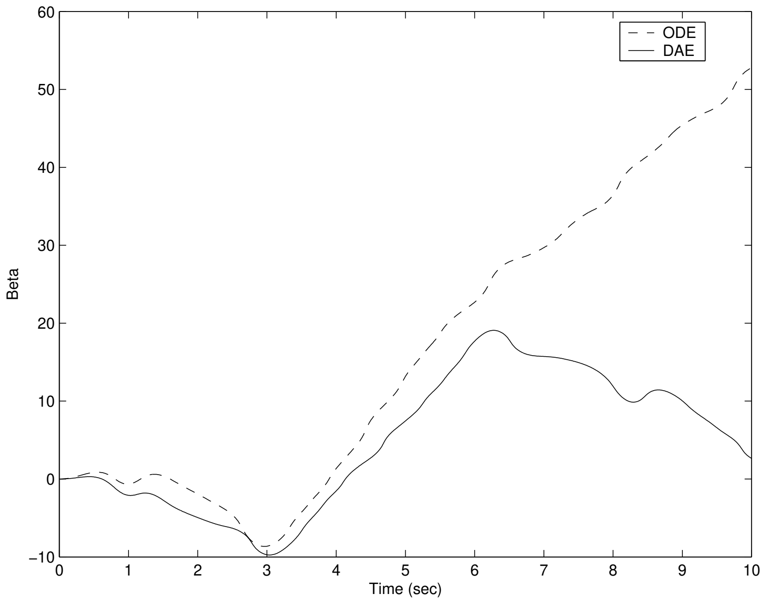

Figure 1.11: Plot of β versus time for the DAE model of the two-link pendulum with Rel. Tol. 10^{-5}, and Abs. Tol. 10^{-6}

solving the ODEs describing the two-link pendulum. The DAE model solved by MBSSIM was also more accurate than the ODE model solved by `ode45`. Unfortunately, it is not possible to conclude that DAE models are more accurate than ODE models because we have only done one comparison. Furthermore, each system is solved using different methods. For further work, it is of interest to model several multibody systems, rigid and flexible, two and three-dimensional, generating equations of motion as ODEs (when possible), index 1,2, and 3 DAEs, and solving them using similar methods.