PENALTY FORMULATION FOR THE NONLINEAR DYNAMIC ANALYSIS OF A DOUBLE PENDULUM

Alem Bazezew
Department of Mechanical Engineering
Addis Ababa University

ABSTRACT

A penalty formulation for the nonlinear analysis of a double pendulum is presented which eliminates the use of Lagrange multipliers method. A numerical algorithm that employs the implicit Runge-Kutta of order 2 along with Newton-Raphson method is developed to analyze the nonlinear dynamics of the system and to study the chaotic behaviour which the system portrays. A computer simulation of a numerical example is given to indicate the effectiveness of the penalty formulation and the algorithm developed.

INTRODUCTION

The simple pendulum, which consists of a point mass m attached to a massless rod of length l, has been extensively studied to analyze nonlinear dynamic behaviour of undamped-free, damped-free, undamped-forced and damped-forced systems [8]. The mathematical model of such systems are nonlinear differential equations. The solution of the equations which describe these systems can be obtained analytically in terms of elliptic integrals for the simpler cases. In general, the equations are integrated numerically to obtain solutions. Numerical results obtained by J.M. Thompson and H.B. Stewart for the simple pendulum are reported in [8].

The other system investigated for nonlinear dynamic behaviour is the planar double pendulum. It consists of two point masses m_1 and m_2 attached to massless rods of lengths l_1 and l_2 moving in a constant gravitational field. This system is used to demonstrate the dynamics of nonlinear systems in mechanics. H.J. Korsch and H.-J. Jodl [6] have studied the double pendulum in which they used independent coordinates φ and ψ to describe the configuration and motion of the system. In introducing these variables, they have used one of the classical approaches of writing the equations of motion. In this approach the equations of motion are derived by using a minimal set of variables which are equal to the number of degrees of freedom of the system. Ensuing second order differential equations of motion of the system have been solved numerically by using three different numerical methods: Euler method, Leapfrog method, and Runge-Kutta method. Their results include trajectories shown in coordinate space, phase space trajectories for the inner and outer pendulum, plots of potential energy and a two-dimensional Poincaré Map.

In this paper, nonlinear dynamic analysis of the double pendulum is discussed in which the configuration and motion of the system is described by natural coordinates which are fully Cartesian. The equations of motion are derived by using Lagrange equations along with constraint equations. As such, the equations of motion obtained are Differential Algebraic Equations (DAE's). These DAE's are then changed to Ordinary Differential Equations (ODE's) by using the Penalty Method which has been described by Bayo *et al* [4]. Numerical solutions to the problem are obtained by using the Mid-Point Rule along with Newton-Raphson method. A numerical algorithm is developed which is then applied to an example. Results obtained are given in various plots and a discussion of the results is also included.

PROBLEM FORMULATION

A double pendulum consisting of two point masses m_1 and m_2 attached to massless rods of length l_1 and l_2, shown in Fig. 1, is considered for nonlinear dynamic analysis. Cartesian coordinates (x,y) are introduced as shown in the Fig. 1 to describe the configuration of the system, i.e. the set of dependent natural coordinates (x_1,y_1) and (x_2,y_2) define the system. Therefore, the vector of position coordinates q which defines the configuration is

$$q = \left[\, x_1 \;\; y_1 \;\; x_2 \;\; y_2 \,\right]^T$$

The coordinates x_1, y_1, x_2, y_2 are related through constraint equations given by Eq. (1).

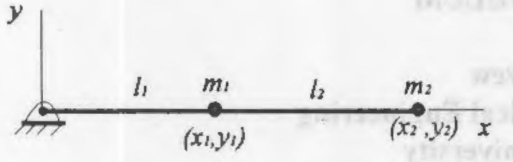

Figure 1 Initial configuration of the double pendulum

$$\Phi_1 = x_1^2 + y_1^2 - l_1^2 = 0$$
$$\Phi_2 = (x_2 - x_1)^2 + (y_2 - y_1)^2 - l_2^2 = 0 \tag{1}$$

EQUATIONS OF MOTION

The system is released from rest in the position shown in Fig. 1. The point masses m_1 and m_2 are subjected to gravitational acceleration g.

With the fully Cartesian dependent coordinates shown in the figure, the kinetic and potential energy of the system, respectively, are:

$$T = \frac{1}{2}m_1(\dot{x}_1^2 + \dot{y}_1^2) + \frac{1}{2}m_1(\dot{x}_2^2 + \dot{y}_2^2) \tag{2}$$

$$V = m_1 g y_1 + m_2 g y_2 \tag{3}$$

One of the most widely used formulations for the dynamic analysis of constrained multi-body systems is the use of Lagrange multipliers method. The equations of motion of the double pendulum are derived from energy considerations and the constraint equations by using Lagrange equations. For the constrained system in consideration, the equation of motion is

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}}\right) - \frac{\partial T}{\partial q} + \frac{\partial V}{\partial q} - \Phi_q^T \lambda = Q_e \tag{4}$$

Substituting for T and V from eqs. (2) and (3), the equation of motion along with constraint equations become

$$M\ddot{q} - \Phi_q^T \lambda + F = 0$$
$$\Phi_1 = x_1^2 + y_1^2 - l_1^2 = 0 \tag{5}$$
$$\Phi_2 = (x_2 - x_1)^2 + (y_2 - y_1)^2 - l_2^2 = 0$$

where M is the mass matrix

$$M = \begin{bmatrix} m_1 & 0 & 0 & 0 \\ 0 & m_1 & 0 & 0 \\ 0 & 0 & m_2 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix}$$

$\ddot{q}$ is the acceleration vector

$$\ddot{q} = [\ddot{x}_1 \;\; \ddot{y}_1 \;\; \ddot{x}_2 \;\; \ddot{y}_2]^T ;$$

λ is a vector of Lagrange multipliers

$$\lambda = [\, \lambda_1 \;\; \lambda_2 \,]^T ;$$

F is the vector of forcing functions

$$F = [\; 0 \;\; -m_1 g \;\; 0 \;\; -m_2 g \;]^T ;$$

and Φ_q is the Jacobian matrix obtained from the vector function of constraint equations and is given by

$$\Phi_q = \frac{\partial \Phi}{\partial q}$$

Equation (5) is a set of differential algebraic equations (DAE's) where the Lagrange multipliers obtained from the constraint equations are part of the variables to be solved for, and the constraint equations are part of the equations to be solved. In other words, the constraints are imposed on the solution variables through the Lagrange multipliers. Solutions to the DAE's are generally complicated mathematical problems. Thus, there is a need for simplified solutions. The penalty formulation is used to simplify the solution procedure to the DAE's by changing them to ODE's. For this purpose, the penalty method is employed whereby the constraint equations are directly incorporated as a dynamical system, penalized by a large factor α, into the equations of motion. The penalty formulation can be regarded as an alternate method to the classical formulations, viz-a-viz:

i) use of the Lagrange multipliers method,
ii) expressing equations of motion in terms of the minimal set of variables, equal in number to the degrees of freedom.

The penalty method is an intuitive method which yields an advantageous form of the equations of motion, especially for singular positions [1]. In this method, a fictitious spring, dashpot and mass are added to the system, in general. For the double pendulum problem, only a fictitious spring or potential is added to the system at the joints between the links and masses as shown in Fig. 2. By so doing, we forget all about the Lagrange multipliers and consider the addition of the potential

$$V^* = \frac{1}{2}\alpha\Phi^2 \tag{6}$$

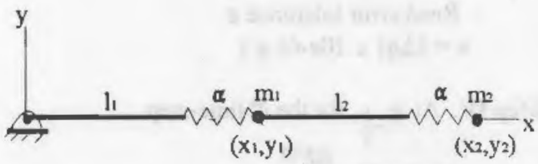

Figure 2 Representation of the double pendulum for penalty formulation

Thus, the constraint equations are introduced into the equation of motion via V^*, and the Lagrange equations given by Eq. (4) become

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}}\right) - \frac{\partial T}{\partial q} + \frac{\partial}{\partial q}(V + V^*) = Q_e \tag{7}$$

where $\frac{\partial V^*}{\partial q} = \alpha\Phi_q\Phi$

The penalty method, like the Lagrange method, operates on the variational formulation of the problem under consideration [2]. But unlike the Lagrange multipliers method, it does not require the introduction of additional unknown variables. However, an important consideration in the method is the choice of the penalty factor α. Therefore, snbstituting for T, V and V^*, and after simplification, the equation of motion becomes

$$M\ddot{q} + \alpha\Phi_q\Phi = Q_e \tag{8}$$

Thus, the set of equations of motion, given by Eq. (5), reduces to a set of ODE's given by Eq. (8). The term $\alpha\Phi_q\Phi$ represents the forces that are generated by the penalty system when the constraints Φ are violated.

In the expanded form, the equation of motion is written as:

$$\begin{bmatrix} m_1 & 0 & 0 & 0 \\ 0 & m_1 & 0 & 0 \\ 0 & 0 & m_2 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{y}_1 \\ \ddot{x}_2 \\ \ddot{y}_2 \end{bmatrix} +$$

$$\alpha \begin{bmatrix} x_1 & (x_1 - x_2) \\ y_1 & (y_1 - y_2) \\ 0 & (x_2 - x_1) \\ 0 & (y_2 - y_1) \end{bmatrix} \begin{bmatrix} \Phi_1 \\ \Phi_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -m_1 g \\ 0 \\ -m_2 g \end{bmatrix}$$

In principle, a fictitious dashpot and mass can also be added to the system along with the fictitious spring. In such case, the equation of motion becomes

$$M\ddot{q} + \alpha\Phi_q[\ddot{\Phi} + \dot{\Phi} + \Phi] = Q_e \tag{9}$$

Including the constraints of velocity and acceleration removes any instability that may arise due to the fictitious spring. The term $\alpha\Phi_q[\ddot{\Phi} + \dot{\Phi} + \Phi]$ represents the forces that are generated when the constraints $\ddot{\Phi}$, $\dot{\Phi}$ and Φ are violated.

The solution of Eq. (8) coincides with the solution of the original problem given by Eq. (5) provided that $\alpha \to \infty$. This condition is achieved by using large penalty factors.

NUMERICAL ALGORITHM TO SOLVE THE EQUATION OF MOTION

The introduction of the penalty factor renders the system of equations to be stiff which requires an unconditionally stable numerical method for its solution. Therefore, Eq. (8) is solved by using the Mid-Point Rule which is an implicit Runge-Kutta method of order 2 and unconditionally stable. The difference equation for the Mid-Point Rule is [5]

$$\begin{aligned} \ddot{q}_{i+\frac{1}{2}} &= \frac{4}{h^2}q_{i+\frac{1}{2}} - \bar{\bar{q}}_i \\ \dot{q}_{i+\frac{1}{2}} &= \frac{2}{h}q_{i+\frac{1}{2}} - \bar{q}_i \end{aligned} \tag{10}$$

where $\bar{\ddot{q}}_i = \frac{4}{h^2}q_i + \bar{\dot{q}}_i$

$\bar{\dot{q}}_i = \frac{2}{h}q_i$, and

h is a pre-set time-step.

These difference equations are used along with the equilibrium equations at $q_{i+\frac{1}{2}}$ to solve the equation of motion. Introducing the difference equations into Eq. (8), the resulting equilibrium equation is obtained to be

$$\frac{4}{h^2}Mq_{i+\frac{1}{2}} + \alpha\Phi_q\Phi - M\bar{\ddot{q}} - Q_e = 0 \qquad (11)$$

Letting

$$F = \frac{4}{h^2}Mq_{i+\frac{1}{2}} + \alpha\Phi_q\Phi - M\bar{\ddot{q}} - Q_e$$

Eq. (11) reduces to

$$F = 0 \qquad (12)$$

This equation is solved for $q_{i+\frac{1}{2}}$ by using the Newton-Raphson method. The method is given by

$$\Delta q = -\left(\frac{\partial F^{(j)}}{\partial q_{i+\frac{1}{2}}}\right)^{-1} F(q^{(j)}_{i+\frac{1}{2}})$$

where $\frac{\partial F}{\partial q_{i+\frac{1}{2}}} = \frac{4}{h^2}M + \alpha\Phi_q{}^T\Phi_q$

Equation (12) is now solved for $q_{i+\frac{1}{2}}$ satisfying a given error limit by iteration. $\ddot{q}_{i+\frac{1}{2}}$ and $\dot{q}_{i+\frac{1}{2}}$ are obtained from Eq.(10). Once $q_{i+\frac{1}{2}}$ is obtained, the half-point value is used to determine q_{i+1} and $\dot{q}_{i+1}$ by using

$$\begin{bmatrix} q_{i+1} \\ \dot{q}_{i+1} \end{bmatrix} = \begin{bmatrix} q_i \\ \dot{q}_i \end{bmatrix} + h \begin{bmatrix} \dot{q}_{i+\frac{1}{2}} \\ \ddot{q}_{i+\frac{1}{2}} \end{bmatrix}$$

A numerical algorithm developed for solving the problem is given below. The calculations in the algorithm are carried out for each time step h.

Step I. -Read initial input data

q_0 , $\dot{q}_0$

-Read number of time-steps n

Step II. Compute $\bar{\ddot{q}}_i$ and $\bar{\dot{q}}_i$ for i^{th} time- step

$$\bar{\ddot{q}}_i = \frac{4}{h^2}q_i + \frac{2}{h}\bar{\dot{q}}_i$$

$$\bar{\dot{q}}_i = \frac{2}{h}q_i$$

Step III. - Initialize $q_{i+\frac{1}{2}}$ for Newton-Raphson iteration

- Read error tolerance ϵ

$\epsilon = \|\Delta q\| \le 10e\text{-}6\| q \|$

Step IV. At $q_{i+\frac{1}{2}}$, for the i^{th} time-step

- compute $\frac{\partial F^{(j)}}{\partial q}$
- compute $F^{(j)}_{i+\frac{1}{2}}$

Step V. At $q^{(j)}_{i+\frac{1}{2}}$:

-Solve for Δq

- Compute $q^{(j+1)}_{i+\frac{1}{2}} = q^{(j)}_{i+\frac{1}{2}} + \Delta q$
- Check for convergence:

If $\Delta q < \epsilon$, go to Step VI

Else, go to Step IV.

Step VI. Compute $\dot{q}_{i+\frac{1}{2}}$ and $\ddot{q}_{i+\frac{1}{2}}$

Step VII. Compute q_{i+1} and $\dot{q}_{i+1}$

Step VIII. Control time-step:

- If number of time-step $I < n$, go to Step I.
- Else, Stop.

NUMERICAL EXAMPLE

A numerical example of the nonlinear dynamic analysis of the double pendulum shown in Figure 1 is solved for

$m_1 = m_2 = 0.2$ kg

$l_1 = l_2 = 1.0$ m

with initial conditions

$$q_0 = [\, x_1 \;\; y_1 \;\; x_2 \;\; y_2 \,]^T = [\, 1 \;\; 0 \;\; 2 \;\; 0 \,]^T$$

$$\dot{q}_0 = [\, \dot{x}_1 \;\; \dot{y}_1 \;\; \dot{x}_2 \;\; \dot{y}_2 \,]^T = [\, 0 \;\; 0 \;\; 0 \;\; 0 \,]^T$$

$$\dot{q}_0 = \begin{bmatrix} \dot{x}_1 & \dot{y}_1 & \dot{x}_2 & \dot{y}_2 \end{bmatrix}^T = \begin{bmatrix} 0 & -g & 0 & -g \end{bmatrix}^T$$

and forcing function

$$Q_e = [\,0\ \ -0.2g\ \ 0\ \ -0.2g\,]^T$$

- The simulation is carried out for t=4s of simulation time using a time step, $\Delta t = h = 0.02s$.
- The penalty factor used is $\alpha = 10e6$.
- The convergence criterion used is $\epsilon = \|\Delta q\| \le 10e\text{-}6\| q \|$

The numerical simulation is carried out using CAL80 commands for nonlinear systems [3].

The results obtained from the computer simulation are shown in Figs. 3-6.

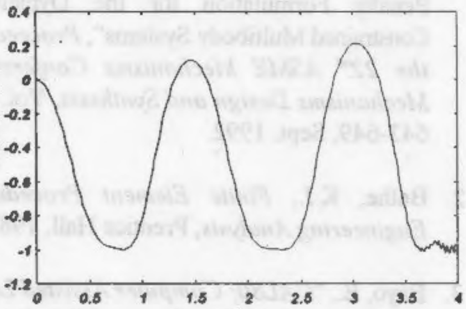

Figure 3 Displacement of mass m_1

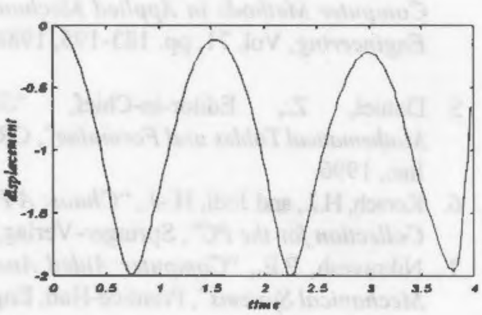

Figure 4 Displacement of mass m_2

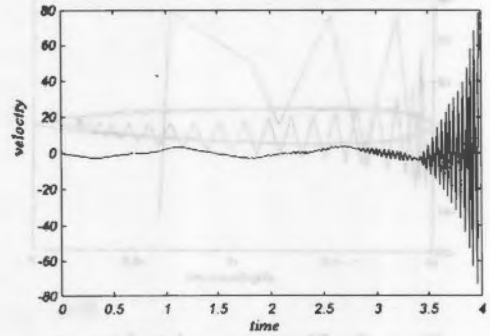

Figure 5 Velocity of mass m_1

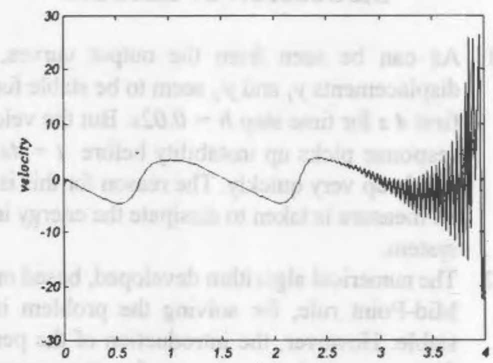

Figure 6 Velocity of mass m_2

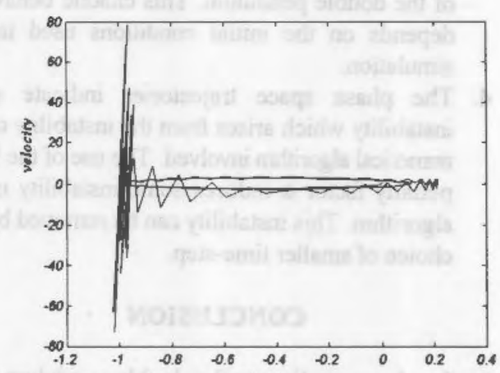

Figure 7 Phase space plot of mass m_1

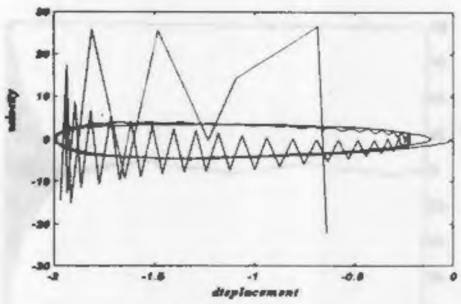

Figure 8 Phase space plot of mass m_2

DISCUSSION OF RESULTS

1. As can be seen from the output curves, the displacements y_1 and y_2 seem to be stable for the first $4\ s$ for time step $h = 0.02s$. But the velocity response picks up instability before $t = 4s$ and builds up very quickly. The reason for this is that no measure is taken to dissipate the energy in the system.
2. The numerical algorithm developed, based on the Mid-Point rule, for solving the problem is A-stable. However, the introduction of the penalty factor α used to account for the constraint equations, causes some instability in the algorithm.
3. Figures 7 and 8 show the phase plots of the chaotic trajectories which show erratic behaviour of the double pendulum. This chaotic behaviour depends on the initial conditions used in the simulation.
4. The phase space trajectories indicate some instability which arises from the instability of the numerical algorithm involved. The use of the large penalty factor α induces some instability in the algorithm. This instability can be removed by the choice of smaller time-step.

CONCLUSION

In the above studies on the double pendulum, new effects may be observed by varying the masses and lengths of the double pendulum. The algorithm developed can be modified to include a fictitious dashpot and mass at the joints. This procedure smoothens the instability in the velocity response.

The Mid-Point Rule used in the numerical algorithm is not unconditionally stable due to the inclusion of the penalty factor. It is stable only for selected time-steps, which need to be worked out. At the same time, the error tolerance should be adequately small to yield good results. With these considerations taken into account, the algorithm developed yields good results and is suitable for the nonlinear analysis of the double pendulum.

Nomenclature

l length
m point mass
M mass matrix
$\boldsymbol{q}$ vector of position coordinates
$\dot{\boldsymbol{q}}$ velocity vector
$\ddot{\boldsymbol{q}}$ acceleration vector
Q_e Vector of forcing functions
T kinetic energy
V potential energy
α penalty factor
$\boldsymbol{\Phi}$ vector function of constrain equations
$\boldsymbol{\Phi}_q$ Jacobian matrix
$\| \|$ vector norm

REFERENCE

1. Avello, A., and Bayo, E., "A Singularity-Free Penalty Formulation for the Dynamics of Constrained Multibody Systems", *Proceedings of the 22nd ASME Mechanisms Conference on Mechanisms Design and Synthesis*, Vol. 46, pp. 643-649, Sept. 1992.

2. Bathe, K.J., *Finite Element Procedures in Engineering Analysis*, Prentice Hall, 1982.

3. Bayo, E., "*CAL80: Computer Assisted Learning of Structural Analysis*", unpublished material, University of California, Santa Barbara, 1988.

4. Bayo, E., Garcia de Jalon, J., and Serna, M.A., "A Modified Lagrangian Formulation for the Dynamic Analysis of Constrained Mechanical Systems", *Computer Methods in Applied Mechanics and Engineering*, Vol. 71, pp. 183-195, 1988.

5. Daniel, Z., Editor-in-Chief, "*Standard Mathematical Tables and Formulae*", CRC Press Inc., 1996.
6. Korsch, H.J., and Jodl, H.-J., "*Chaos: A Program Collection for the PC*", Springer-Verlag, 1994.
7. Nikravesh, P.E., "*Computer Aided Analysis of Mechanical Systems*", Prentice-Hall, Englewood Cliffs, NJ, 1988.
8. Thompson, J.M.T., and Stewart, H.B., "*Nonlinear Dynamics and Chaos*", John Wiley and Sons, 1986.