Numerical simulation and visualization of robotic arm manipulators

H.M. Abdelaziz [a], M.M. Helaly [b] and E.H. Atta [a]

[a] *Informatics Research Institute, Mubarak City for Scientific Research and Technology Applications, Alexandria, Egypt*

[b] *Production Eng. Dept., Faculty of Eng., Alexandria University, Alexandria, Egypt*

An efficient numerical simulation and visualization method has been developed for modeling the dynamics of N-connected bodies such as robotic arm manipulators. The method features Kane-type formulation of the equations of motion, a variable time step integrator, and a visualization procedure for displaying the time history of motion dynamics. Flexibility effects have been incorporated in the developed method by using a novel approximate model of segmented lumped masses and rotational springs. The developed numerical method has been tested on a number of robotic configurations with successful correlation with available data. The test cases include a robotic arm manipulator, and a multi-link flexible beam. The flexibility model has been validated for the case of elastic flat plate wing divergence phenomenon. The divergence speed predicted by the model closely matches the observed experimental data. Correlation of simulation indicates that the developed method is accurate and posses a high level of computational efficiency. Visualization and animation of the numerical simulation results have been affected through combined geometrical modeling and key frame approach. “AVI” type movies have been produced that display the motion time history of the cases considered.

تم تطوير خوارزم وبرنامج عالي الكفاءة للمحاكاة العددية لديناميكا المنظومات المتعددة الأجسام كالأذرع الروبوتية. ويعتمد النموذج الرياضي للخوارزم المطور على معادلات " Kane" للحركة ويشتمل على خوارزم عددي متغير الخطوة لحل معادلات الحركة بالإضافة إلى طريقة لإظهار مسار الحركة الديناميكية في صورة فيلم باستخدام المواصفات "AVI". وقد تم تنفيذ الخوارزم المطور في برنامج باستخدام لغة الفورتران. وقد تم تقييم أداء الخوارزم المطور بالرسالة عن طريق تطبيقه في عدة حالات تضمنت نماذج مختلفة للمنظومات الروبوتية المتعددة الأجسام اشتملت على ذراع روبوت لتحريك الخلايا الشمسية للأقمار الاصطناعية وذراع روبوت للتطبيقات الصناعية وكمرة جسيئة متعددة الأجزاء. وقد دلت مقارنة نتائج البرنامج مع نتائج الأبحاث المنشورة على دقة و كفاءة حسابية عالية. وقد تم تطوير نموذج رياضي لدراسة تأثير مرونة المنظومات المتعددة الأجسام على الأداء الديناميكي يعتمد على نموذج ميكانيكي ذو كتل مركزة وزنبركات ذات الحركة الدائرية. وقد تم تطبيق النموذج لدراسة الديناميكا المرنة الهوائية لجناح مستطيل عند سرعات تحت صوتية و زوايا سقوط متغيرة وقد أظهرت النتائج العددية عند مقارنتها بالنتائج المعملية على إمكانية تحديد سرعة الانفراج الناتجة عن تأثير المرونة على الأداء الايروديناميكى. وقد تم عمل تمثيل ونمذجة لنتائج المحاكاة العددية من خلال تطوير فيلم يوضح التمثيل الزمني لتاريخ الحركة.

Keywords: Kane’s method, Multibody system dynamics, Robotics

1. Introduction

Many common engineering structures, including various types of spacecraft, land vehicles, industrial machinery and robots, can be modeled as multibody systems. The technical literature on the subject is vast, indeed, with publications going back several decades. In some applications multibody structures can be modeled by assuming that all bodies in the structure are rigid, with the derivation of equations of motion carried out by a variety of techniques such as Newton-Euler equations, d’Alembert’s principle, Lagrange’s equations, or the method popularized by Kane. The literature devoted to rigid multibody structures is well established by Amirouche [1], Huston [2-4], Roberson and Schwertassek [5], and Haug [6-7], as well as papers by Kane and Levinson [8-9].

In recent years, there has been a significant rise in the demand for robots to perform increasingly more complex tasks. Dynamic simulation of such systems is an important, if not essential tool used for design, testing, and optimization. The need to examine design variations and alternate system configurations mandates the use of dynamic simulation.

In the last decade, when extensive dynamical studies of multibody spacecraft, robot devices, complex scientific equipment, and biomechanical systems were first undertaken, it became apparent that the straightforward use of classical methods could entail the expenditure of very large, at times even prohibitive amounts of analysts' labor, and could lead to the equations of motion so unwieldy as to render computer solutions unacceptably slow for technical and/or economical reasons. During the past decade, a number of methods of dynamical analysis have been developed and employed to overcome this difficulty. Efforts were undertaken to reduce the formulations of the equations of motion for complex systems to a truly simpler task. A great deal of effort has been involved in the development of multibody computer programs, programs intended to generate and solve equations of motion simultaneously for user-specified arrangements of connected bodies. A leading and increasingly popular method in multibody dynamics is based on Kane's equations (or Lagrange's form of d'alembert's principle, or principle of virtual power) [9-10] for obtaining the governing equations of motion. The use of Kane's equations leads to major saving in labor as well as to simpler equations of motion. Moreover being highly systematic, this method focus attention on motions rather than on configurations, giving the analyst maximum physical insight not involving variations, such as those encountered in connection with virtual work, it can be presented at a relatively elementary mathematical level. Furthermore, for multibody systems kane's equations have been shown to possess the advantages of both Lagrange's equations and Newton-Euler methods but without the corresponding advantages. That eliminated from the analysis enables one to deal directly with non-holonomic systems without the introduction of scalar energy function and their derivatives. Also kane's equations are ideally suited for accommodating generalized speeds and quasi-coordinates. The equations may be derived in forms that are easily converted into numerical algorithms.

2. Method formulation

The present method is based on Kane's formulation of the equations of motion. This formulation introduces the new concept of generalized speeds that can add up to characterize the speed of the entire system. By focusing on generalized speeds instead of Newtonian or Lagrangian coordinates, the equations of motion produced are far simpler than those of the classical methods, especially for complex multibody problems.

2.1. Formulation of Kane's equations of motion

The formulation of the equations starts by defining the generalized speeds, partial angular velocities, partial velocities; these are followed by the definition of the generalized inertia forces, and the generalized active forces. The expression of these quantities is given below:

1) Generalized speeds;

$$u_i = \sum_{j=1}^{n} w_{ij} \overset{\circ}{q}_j + x_i \quad (i = 1,\ldots n). \tag{1}$$

Where w_{ij}, x_i are functions of $q_1,\ldots,q_n$, t and can solve uniquely for $q_1^o,\ldots, q_n^o$

2)-a. Partial angular velocity;

$$\omega = \sum_{r=1}^{n} \omega_r u_r + \omega_t, \tag{2}$$

where, ω is the angular velocity of the rigid body (rad/s), ω_r is called partial angular velocity (rad/s), ω_r, ω_t are functions of $q_1,\ldots ,q_n$ and the time t.

2)-b. partial linear velocity;

$$V = \sum_{r=1}^{n} V_r u_r + V_t \quad , \tag{3}$$

where, V. is the linear velocity of a point in the rigid body (m/s), V_r is called partial linear velocity (m/s), V_r, V_t are functions of $q_1,\ldots,q_n$ and the time t.

3)-a. Generalized inertia forces/torques;

The generalized inertia forces/toques are defined as:

$$(F_r^*)_B = {}^N\omega_r^B . T^* + {}^N V_r^B . F^* \qquad (r = 1,\ldots,n), \tag{4}$$

where N is the inertial reference frame, n is number of the degrees of freedom for the body under consideration, ${}^N\omega_r^B$ is the partial angular velocity for body B, and ${}^N V_r^B$ is the partial linear velocity for body B.

The terms T^* and F^* are defined as follows:

$$T^* = (I^{B/B^*} . {}^N\omega^B) \times {}^N\omega^B - I^{B/B^*} . {}^N\alpha^B \quad \text{(N.m)}. \tag{5}$$

$$F^* = -m_B \, {}^N a^{B^*} \quad \text{(N)}. \tag{6}$$

I^{B/B^*} is the moment of inertia for body B (Kg.m^2),

${}^N\omega^B$ is the angular velocity for body B with respect to inertial reference frame (rad/s),

${}^N\alpha^B$ is the angular acceleration for body B with respect to inertial reference frame (rad/s^2), and

${}^N\alpha^{B^*}$ is the linear acceleration for point B^* (the C.G. of the body B) with respect to inertial reference frame (m/s^2).

3)-b. generalized active forces and torques;

The generalized inertia forces/toques are defined as:

$$(F_r)_B = {}^N\omega_r^B . T + {}^N V_r^P . F \quad (r = 1,\ldots,n) \text{ (N)}, \tag{7}$$

as p is the point of action of the force acting the body.

The equations of motion then takes the simple form of:

$$F_r^* + F_r = 0 \qquad (r = 1,\ldots,n). \tag{8}$$

Details of method implementation for a configuration of four connected bodies are presented in appendix A. The equations of motions are integrated using the efficient Kutta-Merson algorithm with a variable time step for stability considerations. Inversion of the matrices is accomplished by using LU decomposition method.

2.2. Visualization and animation of numerical results

The visualization and animation of the computed dynamics results are effected in two-step procedure. The first step involves the creation of a three dimensional geometrical model that represents the robotic system under consideration. In the second step the position of each link of the robotic system is established from the numerical results that describe the time history of the motion dynamics. A number of frames are thus created with each frame representing the position of the robotic system at a specified time interval.

The procedure described above is implemented by using the software 3D studio max [11]. The software offers the capability to create geometrical models and AVI movies composed of frames supplied by the user. Other features include the addition of different textures and light modes to enhance the appearance of the developed model. It also provides the flexibility of specifying camera position enabling model viewing from different directions. The following is a brief outline of the steps taken to develop an AVI movie for the motion dynamics resulting from the numerical simulation of the of a given robotic system.

A screen shot showing a robotic model displayed on 3D studio max screen is shown in fig. 1.

2.3. Numerical simulation results

The present algorithm is validated by performing numerical motion simulation for a number of test cases and comparing the results with available data. Visualization and animation of the time history of the motion has been also performed by constructing geometric models of the cases under consideration and feeding the computed results into a number of frames that represents the instantaneous position in each case.

3. Test cases

3.1. Robotic arm manipulator

The first case considered represents a robotic arm manipulator of ref. [10].

Fig. 2 displays a schematic representation of the robot arm consisting of three elements A, B and C, the last of which holds a rigid body D rigidly. One end of A is a hub that is made to rotate about a vertical axis fixed in the reference frame N. B is connected to A at a point P by means of a motor (all parts of which are rigidly attached either to A or to B) that causes B to rotate relative to A about a horizontal axis fixed in A passing through P, and perpendicular to the axis of A. finally, C is connected to B by means of a rack-and-pinion drive that can make C slide relative to B.

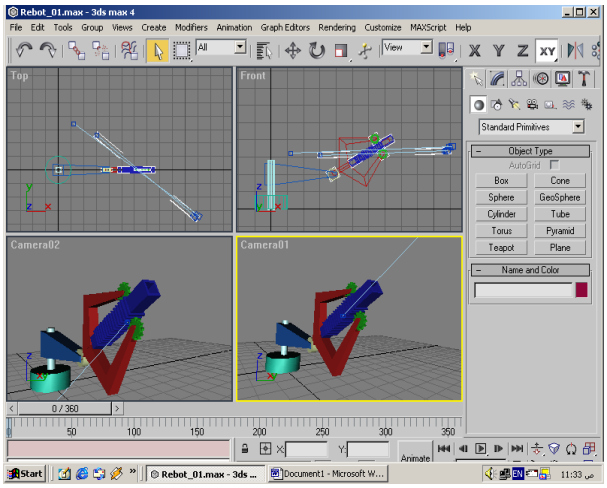


Fig. 1. Graphical interface showing model being developed.

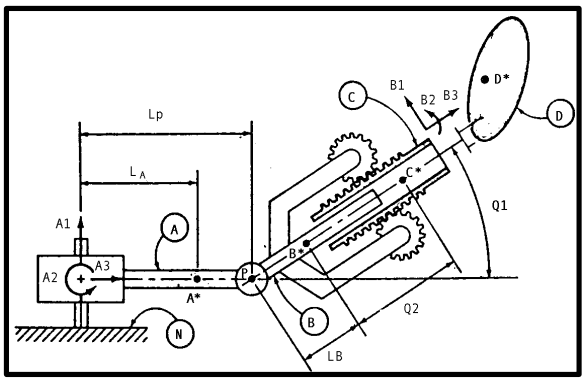


Fig. 2. Schematic representation of robot arm.

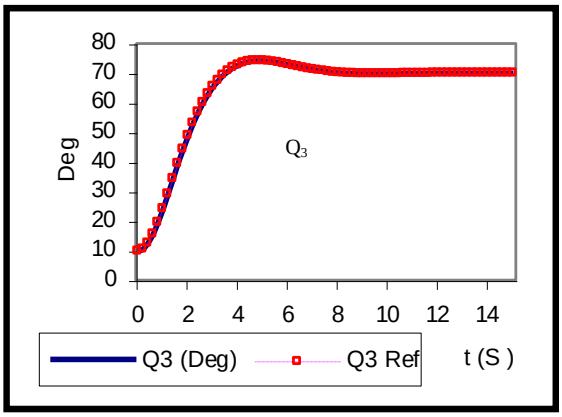


Fig. 3. Motion history simulation of manipulator arm position angle Q3.

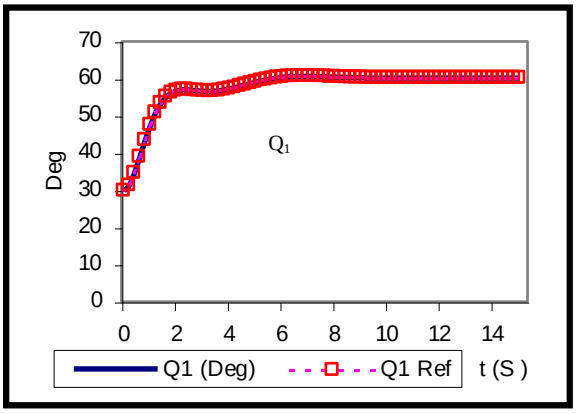


Fig. 4. Motion history simulation of manipulator arm position angle Q1.

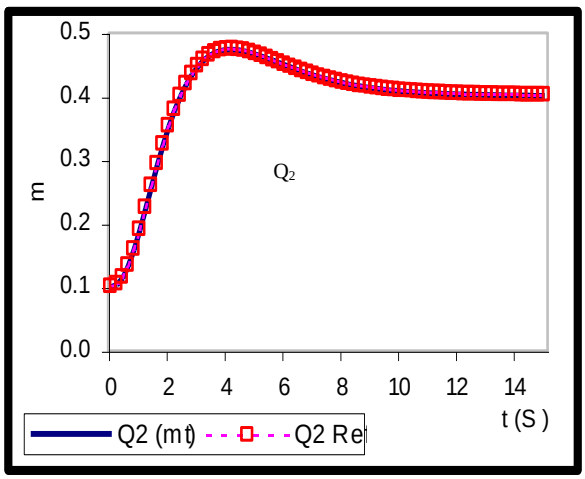


Fig. 5. Motion history simulation of manipulator arm position angle Q2.

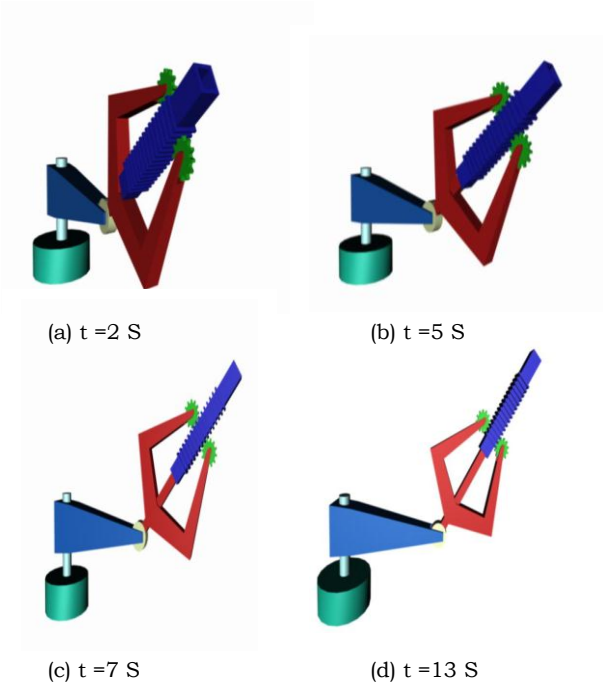


Fig. 6. Sample frames of motion animation.

The robot arm considered has three degrees of freedom with generalized speeds (U_1, U_2, U_3), external forces and torques are given above. The data required to perform the numerical simulation are taken from ref. [10]. Motion simulation results are displayed in figs. 3, 4 and 5 along with the results of ref. [10] for a duration of 15 seconds. The results indicate a rapid increase, a slight overshoot followed by a steady value for the position angles Q_3, Q_1, and displacement Q_2, which accurately correlate with the results of ref. [10].

3.1.1. Visualization of simulation results

The computed results of motion simulation are visualized and animated using a constructed geometric model and a number of key frames representing the time history of the motion dynamics in an animated fashion. Samples of key frames at different times are displayed in fig. 6.

3.2. Flexible multi-link beam model

In this case a multi-link beam model having four beams of square cross section and rotational springs between adjacent links fig. 7 is used to model the elastic divergence of a planar aircraft wing. Wing divergence occurs when a wing deflects under aerodynamic load so as to increase the applied load, or move the load so that the twisting effect on the structure is increased. The increased load deflects the structure further, which causes a further increase in load, until the structure fails. This case is selected to test the capabilities of the present algorithm in handling elastic effects.

Aerodynamic forces and moments are first computed using a linear unsteady vortex-Lattice method, the details of which can be found in [12] to provide the external forces and moments needed in solving the dynamical equations of motion of the multi beam link. The stiffness (K) of the rotational springs constants are computed using the approximate value $K = EI/L$ that is based on the static deflection of a cantilever beam, where I is the beam moment of inertia, E is the modulus of elasticity, and L is the beam length.

At each time step the aerodynamic forces which are function of the links deflections are computed and then used as external applied force at each beam link representing the planer wing. The deformation of each link is then computed by solving dynamical equations of motion of the multi beam links resulting in new link deflections that will cause the aerodynamic forces to change.

The wing model selected is known to have a divergence speed at about 110 m/s, the data used in the computation are as follows: Beam length L=0.085725m, Beam width B= 0.001455 m, Beam mass m=0.124 kg, Modulus of elasticity $E = 2 \times 10^{11}$ N/m^2.

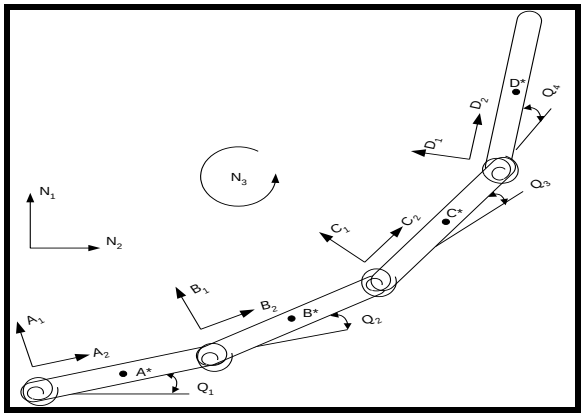
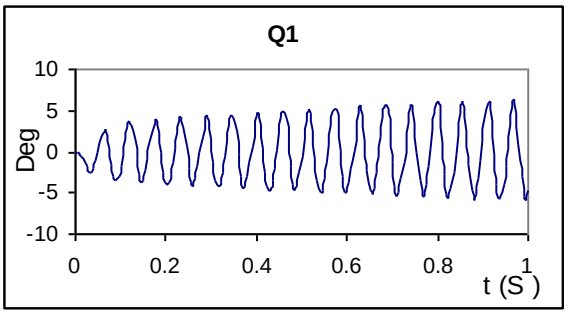
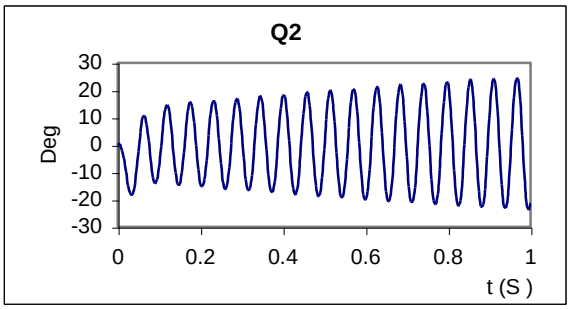


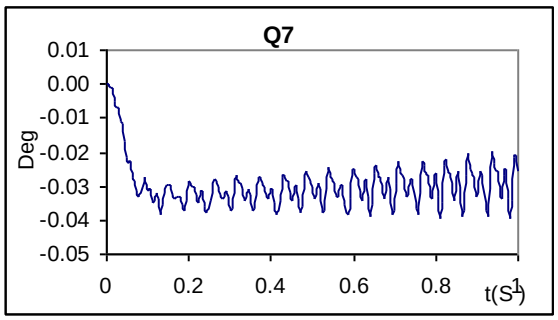
Fig. 7. Flexible multi-link beam model.



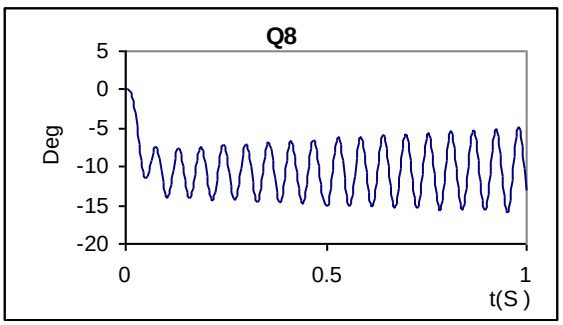
(a) bending mode



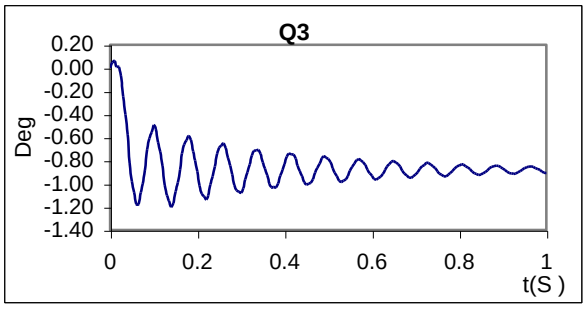
(b) torsion mode



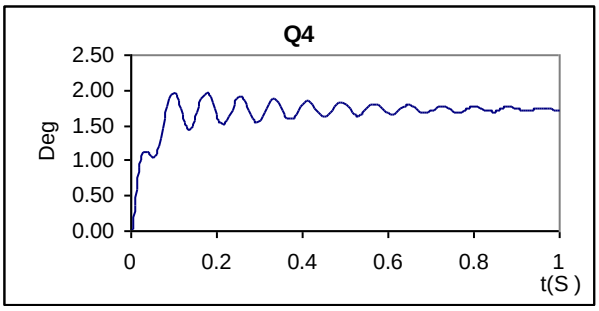
(c) bending mode



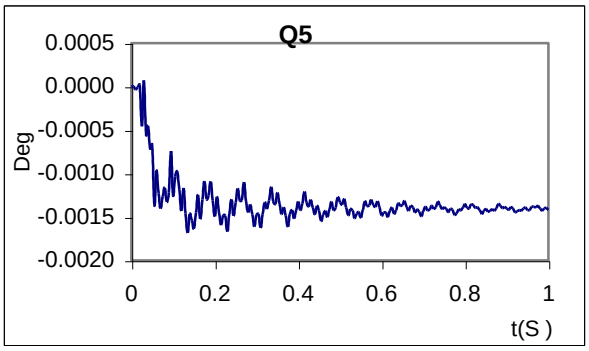
(d) torsion mode



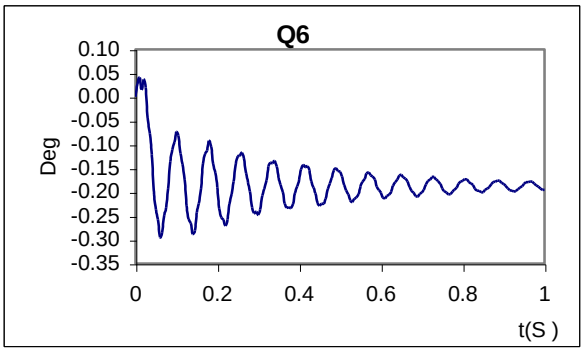
(e) Bending mode



(f) Torsion mode



(g)Bending mode



(h) Torsion mode

Fig. 8. Variation of deflection angles of the flexible multi link beam wing model at an air speed a, b, c, d Air speed = 100m/s close to the divergence speed e, f, g, h Air speed = 60m/s below the divergence speed.

Numerical simulation is performed for two air speeds of 60 and 100 m/s respectively at an angle of incidence of one degree and duration of about one second. Simulation results are displayed in fig. 8 for the torsional and bending deflection modes. The results clearly show a diverging oscillations and instability at air speed of 100 m/s close to the divergence speed in contrast to the results at 60 m/s that exhibit damped oscillations. The computational results demonstrate the capability of the present model in approximately addressing the effects of elastic effects in dynamical systems.

4. Conclusions and recommendations

In this paper the problem of numerical simulation of the dynamics of multi body systems exemplified in robotic arm manipulator is addressed. An efficient formulation of the equations of motion has been developed based on Kane's method for dynamics modeling, general enough to handle n-connected bodies. The equations of motion are integrated using a fast algorithm utilizing LU decomposition and a variable time step Kutta-Merson method. The developed numerical algorithm has been implemented in a robust FORTRAN code that outputs the time motion history for the case under consideration.

Code validation is implemented by performing numerical simulations for several test cases including robotic arm manipulator, and a flexible multiconnected beam. Correlation of simulation results with available data indicates that the developed method is accurate and posses a high level of computational efficiency.

Flexibility effects have been incorporated in the developed model in an approximate fashion utilizing a segmented lumped mass approach and rotational springs. The flexibility model has been validated for the case of elastic flat plate wing divergence phenomenon. The divergence speed predicted by the model closely matches the observed experimental data.

The visualization and animation of the numerical simulation results are effected in a two-step procedure. The first step involves the creation of a three dimensional geometrical model that represents the system under consideration. In the second step the position angles describing the orientation of the system are established from the numerical results that describe the time history of the motion dynamics. A number of frames are thus created with each frame representing the position of the system at a specified time interval. The animation procedure is then finalized by stacking the frames to produce an AVI movie. Correlation results and the visualization movies demonstrate the effectiveness of the developed method in performing accurate dynamic simulation and visualization for robotic arm manipulators and general multi-connected bodies.

Further improvements to the developed numerical procedure could still be added; these could include improving the treatment of flexibility effects using more accurate modeling such as finite element techniques and including motion constraints consideration in the dynamic formulation

Appendix A

Formulation of the equations of motion for a multibody configuration consisting of four connected objects:

Define generalized speeds:

$^{N}V^{A^*}=U_aE_a$,

$^{N}\omega^{A}=U_aE_a$,

$^{N}\omega^{B}=U_bE_b$,

$^{N}\omega^{C}=U_cE_c$,

$^{N}\omega^{D}=U_dE_d$

Define length vector:

$^{A^*}r^{P1}=L_a{}^{+}E_a$ $^{P1}r^{B^*}=L_b{}^{-}E_b$

$^{B^*}r^{P2}=L_b{}^{+}E_b$ $^{P1}r^{P2}=L_bE_b$

$^{P2}r^{C^*}=L_c{}^{-}E_c$ $^{C^*}r^{P3}=L_c{}^{+}E_c$

$^{P2}r^{P3}=L_cE_c$ $^{P3}r^{D^*}=L_d{}^{-}E_d$,

as E_a unit vectors define each body local axis:
Velocity of CG'S (Translation Velocity):

$^{N}V^{A^*}= U_aE_a,$ $^{N}V^{B^*}= [U_aE_a+ (U_a\times L_a{}^{+})E_a] + [(U_b\times L_b{}^{-})E_b]$

$^{N}V^{C^*}= [U_aE_a+ (U_a\times L_a{}^{+})E_a + (U_b\times L_b{}^{-})E_b] + [(U_c\times L_c{}^{-})E_c]$

$^{N}V^{D^*}= [U_aE_a+ (U_a\times L_a{}^{+})E_a + (U_b\times L_b{}^{-})E_b + (U_c\times L_c{}^{-})E_c]$
$+ [(U_d\times L_d{}^{-})E_d]$

Coordinate system transformations:

$E_b=E_aR_{ab},$ $E_c=E_aR_{ac}$, $E_d=E_aR_{ad}$

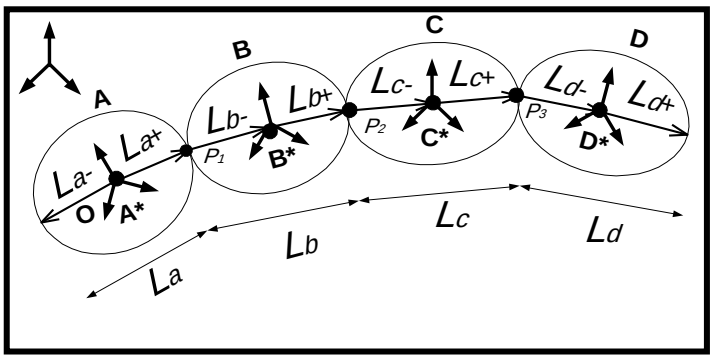


Fig. 9. Four body system.

Partial velocity

$$U=\begin{bmatrix} U_1 \\ U_2 \\ . \\ . \\ U_n \end{bmatrix},\ V=\begin{bmatrix} V_1 \\ V_2 \\ . \\ . \\ V_m \end{bmatrix},\ \frac{\partial U}{\partial V}=\begin{bmatrix} \frac{\partial U_1}{\partial V_1} & \frac{\partial U_1}{\partial V_2} & . & . & \frac{\partial U_1}{\partial V_m} \\ \frac{\partial U_2}{\partial V_1} & \frac{\partial U_2}{\partial V_2} & . & . & . \\ . & . & . & . & . \\ \frac{\partial U_n}{\partial V_1} & \frac{\partial U_n}{\partial V_n} & . & . & \frac{\partial U_n}{\partial V_m} \end{bmatrix},$$

$$\text{define } \nabla \times L=\begin{bmatrix} 0 & L_3 & -L_2 \\ -L_3 & 0 & L_1 \\ L_2 & L_1 & 0 \end{bmatrix}$$

$$V_r=\begin{bmatrix} Ea & 0 & 0 & 0 & 0 \\ Ea & Ea\nabla \times La^{+} & EaRab\nabla \times Lb^{-} & 0 & 0 \\ Ea & Ea\nabla \times La^{+} & EaRab\nabla \times Lb & EaRac\nabla \times Lc^{-} & 0 \\ Ea & Ea\nabla \times La^{+} & EaRab\nabla \times Lb & EaRac\nabla \times Lc & EaRad\nabla \times Ld \end{bmatrix},$$

as $La+$ and $La-$ is the distance of any C.G. and consecutive point.

Acceleration:

$^{N}a^{A*} = Ea(U'o+Ua\times Uo),$

$^{N}a^{B*} = Ea(U'o + Ua\times Uo + U'a\times La^{+} + Ua \times Ua\times La^{+})) + Eb(U'_b\times L_b^{-} + U_b \times (U_b\times L_b^{-}))$

$^{N}a^{C*} = Ea(U'o + Ua\times Uo + U'a\times La^{+} + Ua \times (Ua\times La^{+})) + Eb(U'_b\times L_b^{-} + U_b \times (U_b\times L_b^{-})) + E_c(U'_c\times L_c^{-} + U_c \times (U_c\times L_c^{-}))$

$^{N}a^{D*} = Ea(U'o + Ua\times Uo + U'a\times La^{+} + Ua \times (Ua\times La^{+})) + Eb(U'_b\times L_b^{-} + U_b \times (U_b\times L_b^{-})) + E_c(U'_c\times L_c^{-} + U_c \times (U_c\times L_c^{-}))$
$+ E_d(U'_d\times L_d^{-} + U_d \times (U_d\times L_d^{-}))$

$$\begin{bmatrix} a^{A*} \\ a^{B*} \\ a^{C*} \\ a^{D*} \end{bmatrix} = Ea \begin{bmatrix} a^{a} \\ a^{b} \\ a^{c} \\ a^{d} \end{bmatrix}.$$

Generalized inertia force F^*_f (force part):

$$F_f^* = -(V_r)^T \begin{bmatrix} m_a a^{A*} \\ m_b a^{B*} \\ m_c a^{C*} \\ m_d a^{D*} \end{bmatrix} = -\begin{bmatrix} I & I & I & I \\ 0 & (\nabla \times La^+)^T & (\nabla \times La^+)^T & (\nabla \times La^+)^T \\ 0 & (R_{ab} \times L_b^-)^T & (R_{ab} \times L_b)^T & (R_{ab} \times L_b)^T \\ 0 & 0 & (R_{ac} \times L_c^-)^T & (R_{ac} \times L_c)^T \\ 0 & 0 & 0 & (R_{ad} \times L_d^-)^T \end{bmatrix} \begin{bmatrix} m_a a^a \\ m_b a^b \\ m_c a^c \\ m_d a^d \end{bmatrix}$$

Inertia torque:

T^{A*}= $-E_a I_a U'_a - E_a (U_a \times I_a U_a)$
T^{B*}= $-E_b I_b U'_b - E_b (U_b \times I_b U_b)$
T^{C*}= $-E_c I_c U'_c - E_c (U_c \times I_c U_c)$
T^{D*}= $-E_d I_d U'_d - E_d (U_d \times I_d U_d)$

partial angular velocity:

$$\omega_r = \begin{bmatrix} 0 & E_a & 0 & 0 & 0 \\ 0 & 0 & E_b & 0 & 0 \\ 0 & 0 & 0 & E_c & 0 \\ 0 & 0 & 0 & 0 & E_d \end{bmatrix}$$

Generalized inertia force F_T^* (torque part):

$$F_T^* = (\omega_r)^T \begin{bmatrix} T^{A*} \\ T^{B*} \\ T^{c*} \\ T^{D*} \end{bmatrix} = -\begin{bmatrix} 0 \\ I_a U_a' + U_a \times I_a U_a \\ I_b U_b' + U_b \times I_b U_b \\ I_c U_c' + U_c \times I_c U_c \\ I_d U_d' + U_d \times I_d U_d \end{bmatrix}$$

Generalized active force Fr:

Given m_a, m_b, m_c, m_d, I_a, I_b, I_c, I_d, R_{ab}, R_{ac}, R_{ad}, L_{a^+}, L_{b^-}, L_{b^+}, L_{c^-}, L_{c^+}, L_{d^-},
At each instant when U_o, U_a, U_b, U_c, U_d are known.

$$F_r = AU' + B$$

Where $U' = \begin{bmatrix} U'_o \\ U'_a \\ U'_b \\ U'_c \\ U'_d \end{bmatrix}$, $A = \begin{bmatrix} A(1,1) & . & . & . & A(1,5) \\ . & & & & . \\ . & & & & . \\ . & & & & . \\ A(5,1) & . & . & . & A(5,5) \end{bmatrix}$, $B = \begin{bmatrix} B(1) \\ B(2) \\ B(3) \\ B(4) \\ B(5) \end{bmatrix}$,

and

$A(1,1) = -(m_a + m_b + m_c + m_d)$
$A(1,2) = -(m_b + m_c + m_d)\ (\nabla \times L_{a+})$
$A(1,3) = -R_{ab}(m_b \nabla \times L_b - + m_c \nabla \times L_b + m_d \nabla \times L_b)$
$A(1,4) = -R_{ac}(m_c \nabla \times L_c - + m_d \nabla \times L_c)$
$A(1,5) = -R_{ad}(m_d \nabla \times L_{d-})$
$A(2,1) = -(\nabla \times L_{a+})^T (m_b + m_c + m_d) = [A(1,2)]^T$
$A(2,2) = -I_a - (\nabla \times L_{a+})^T (m_b + m_c + m_d)\ (\nabla \times L_{a+})$
$A(2,3) = -(\nabla \times L_{a+})^T R_{ab}(m_b \nabla \times L_b - + m_c \nabla \times L_b + m_d \nabla \times L_b)$
$A(2,4) = -(\nabla \times L_{a+})^T R_{ac}(m_c \nabla \times L_c - + m_d \nabla \times L_c)$
$A(2,5) = -(\nabla \times L_{a+})^T R_{ad}(m_d \nabla \times L_{d-})$
$A(3,1) = -(m_b \nabla \times L_{b-} + m_c \nabla \times L_b + m_d \nabla \times L_b)^T R^T_{ab} = [A(1,2)]^T$
$A(3,2) = -(m_b \nabla \times L_{b-} + m_c \nabla \times L_b + m_d \nabla \times L_b)^T R^T_{ab} (\nabla \times L_{a+}) = [A(2,3)]^T$
$A(3,3) = -I_b - (\nabla \times L_{b-})^T m_b(\nabla \times L_{b-}) - (\nabla \times L_b)^T (m_c + m_d)(\nabla \times L_b)$
$A(3,4) = -(\nabla \times L_b)^T R^T_{ab} R_{ac}(m_c \nabla L_c - + m_d \nabla \times L_c)$
$A(3,5) = -(\nabla \times L_b)^T R^T_{ab} R_{ad}(m_d \nabla \times L_{d-})$
$A(4,1) = -(m_c \nabla \times L_{c-} + m_c \nabla \times L_b + m_d \nabla \times L_b)^T R^T_{ab} = [A(1,4)]^T$
$A(4,2) = -(m_c \nabla \times L_{c-} + m_d \nabla \times L_c)^T R^T_{ac} (\nabla \times L_{a+}) = [A(2,4)]^T$
$A(4,3) = -(m_c \nabla \times L_{c-} + m_d \nabla \times L_c)^T R^T_{ac} R_{ab} (\nabla \times L_b) = [A(3,4)]^T$

$A(4,4)= -I_c-(\nabla\times L_c-)^T\ m_c(\nabla\times L_c-) - (\nabla\times L_c)^T\ m_d\ (\nabla\times L_c)$

$A(4,5)= -\ (\nabla\times L_c)^T\ R^T_{ac}\ m_d\ R_{ad}\ \nabla\times L_d-$

$A(5,1)= -(\nabla\times L_d-)\ R^T_{ad}\ \ m_d=[A(1,5)]^T$

$A(5,2)= -\ (\nabla\times L_d-)^T\ R^T_{ad}\ m_d\ \nabla\times L_{a+}=[A(2,5)]^T$

$A(5,3)= -\ (\nabla\times L_d-)^T\ R^T_{ad}\ m_d\ R_{ab}\ \nabla\times L_b=[A(3,5)]^T$

$A(5,4)= -\ (\nabla\times L_d-)^T\ R^T_{ad}\ m_d\ R_{ac}\ \nabla\times L_c=[A(4,5)]^T$

$A(5,5)= -I_d-(\nabla\times L_d-)^T\ m_d(\nabla\times L_d-)$

$T_a= (U_a\times U_o)m_a$

$T_b= [U_a\times U_o +U_a\times(U_a\times L_{a+})+R_{ab}\ U_b\ \times(U_a\times L_b-)]m_b$

$T_c= [U_a\times U_o +U_a\times(U_a\times L_{a+})+R_{ab}\ U_b\ \times(U_a\times L_b)\ +R_{ac}\ U_c\ \times(U_c\times L_c-)]m_c$

$T_d= [U_a\times U_o +U_a\times(U_a\times L_{a+})+R_{ab}\ U_b\ \times(U_a\times L_b)\ +R_{ac}\ U_c\ \times(U_c\times L_c)+\ R_{ad}\ U_d\ \times(U_d\times L_d-)]m_d$

$B(1)=-(T_a+T_b+T_c+T_d)$

$B(2)=-\ U_a\times I_aU_a-(\nabla\times L_{a+})^T\ (T_b+T_c+T_d)$

$B(3)=-\ U_b\times I_bU_b-[\ (\nabla\times L_b-)^T\ R^T_{ab}T_b+\ (\nabla\times L_b\ )^TR^T_{ab}\ (T_c+T_d)]$

$B(4)=-\ U_c\times I_cU_c-[\ (\nabla\times L_c-)^T\ R^T_{ac}T_c+\ (\nabla\times L_c\ )^TR^T_{ac}\ T_d]$

$B(5)=-\ U_d\times I_dU_d-(\nabla\times L_d-)^T\ R^T_{ad}\ T_d.$

References

[1] M.L. Farid. Amirouche., Computational Methods in Multibody Dynamics, Prentice-Hall international, inc. (1992).

[2] R.L. Huston and C.E. Passerllo and M. W. Harlow, Dynamics of Multi-Rigid-Body Systems, J. Apple. Mech, Vol. 45 (4), pp. 889-894 (1978).

[3] R.L. Huston and C.E. Passerllo, "On Multi-Rigid-Body Systems Dynamics", Comput. Struct., Vol. 10, pp.439-446 (1979).

[4] R.L. Huston and C.E. Passerllo, "Multibody Dynamics Including Translation Between The Bodies", Comput. Struct., Vol. 11, pp.713-720 (1980).

[5] R. Shwertassek and R.E. Robenson, "A State Space Dynamical Representation for Multibody Mechanical Systems, Part I: Systems with Tree Configuration", Acta Mech., Vol. 50, pp. 141-161 (1984).

[6] S.S. Kim and E.J. Haug, "A Recursive Formulation for Flexible Multibody Dynamics, Part I: Open-Loop Systems," Computer Methods in Applied Mechanics and Engineering, Vol. 71, pp. 293-314 (1988).

[7] S.S. Kim and E.J. Haug "A Recursive Formulation for Flexible Multibody Dynamics, Part II: Closed-Loop Systems," Computer Methods in Applied Mechanics and Engineering, Vol. 74, pp. 251-269 (1989).

[8] T.R. Kane and D.A. Levinson, "The Use of Kane's Dynamical Equations in Robotics," The International Journal of Robotics Research, Vol. 2 (3), pp. 3-21 (1983).

[9] T.R. Kane and C.F. Wang, on the Derivation of Equations of Motion, J. Soc. lnd. App. Math., Vol. 13 (1969).

[10] T.R. Kane and D.A. Levinson, Dynamics: Theory and Applications, McGraw- Hill, New York, (1985).

[11] 3D Studio Max, Version 3, NewRiders Publishing and its Licensors Copyright © (1997).

[12] P. Stephen, Timoshenko and J.N. Goodier Theory of Elasticity, McGraw Hill College Div; 3rd edition December (1970).

Received April 22, 2004
Accepted June 28, 2004