
The theory and application of a classical Eulerian multibody code

D. I. M. Forehand^a (corresponding author), M. P. Cartmell and R. Khanin^b

^a *School of Engineering, Robert Gordon University, School Hill, Aberdeen AB10 1FR, UK*

E-mail: D.I.M.Forehand@rgu.ac.uk

^b *Department of Statistics, University of Glasgow, Glasgow, G12 8QQ, UK*

Abstract The topic of multibody analysis deals with the automatic generation and subsequent solution of the equations of motion for a system of interconnected bodies. Many academic and industrial computer programs have been developed to carry out this task. However, although some of these programs can obtain the equations of motion in *fully* symbolic form, it is believed that all the existing programs for multibody analysis solve these equations numerically. The idea behind the research which underpins this paper is to select and implement a *symbolic* version of an existing multibody algorithm and then to integrate it with a recently developed solver which can obtain approximate *analytical* solutions to the equations of motion. The present paper deals with the selection of this multibody method. The theory behind the chosen method (the Roberson and Schwertassek algorithm) is described in some detail but also in a much more concise form than can be found in the literature. In addition, a *fully* worked example of the application of the multibody algorithm to a practical physical problem is given. Such examples are rare in the literature, and so it is intended that this paper can serve as a basis for enhanced didactic practice in this traditionally difficult subject area.

Keywords multibody analysis; Roberson and Schwertassek method; Eulerian approach; graph theory; relative or joint coordinates; symbolic dynamic analysis

Introduction

Over the last few decades there has been a great deal of research into the dynamics of multibody systems. As an abstraction, a multibody system is a collection of bodies which are connected together by a series of joints. In order to analyse the dynamics of such systems a number of computer algorithms have been developed and these algorithms are usually referred to as ‘multibody codes’. Given only a brief description of a multibody system as input, a multibody code will automatically derive its equations of motion. Then, in most cases, the multibody code will go on to solve these equations to get the response of the system.

A number of academic and commercial multibody codes have been developed and in the main these codes fall into one of two groups.

The first group are called ‘Eulerian’ codes because they use the Newton/Euler equations of motion as a direct starting point. Examples of Eulerian codes are: NEWEUL by Schiehlen and Kreuzer [1, 2], RESTRI/PLANET by Richard and Andrews (see [3, 4]), and MESA-VERDE by Wittenburg [5, 6].

The second group are called ‘Lagrangian’ codes. These use results from the field of analytical mechanics, such as Lagrange’s or Hamilton’s equations, as their direct starting point. Examples of Lagrangian codes are: ADAMS (Automatic Dynamic Analysis of Mechanical Systems) by Orlandea (see [7, 8]), DADS (Dynamic

Analysis and Design System) by Haug (see [9]), and IMP (Integrated Mechanisms Program) by Uicker (see [10]).

Within these two approaches, the number of equations used to describe the motion of the system varies from a minimum number of purely differential equations (the *state-space* equations of motion) to a maximum set of differential algebraic equations (DAEs). The minimum set is usually coupled, non-linear and complicated in nature, whereas the maximum set includes algebraic constraint equations which are simple in form and have a sparse coefficient matrix. For a certain class of multibody system, Wittenburg's formulation [5] produces a minimal set of equations. In contrast, Orlandea *et al.*'s algorithm [7] produces a very large set of equations, which are then solved with the help of sparse matrix methods.

Another characteristic feature of a multibody code is the type of coordinates that it uses. *Absolute* coordinates relate the position and orientation of a body in a multibody system to the main reference frame. *Relative* (or *joint*) coordinates, on the other hand, relate the relative position and orientation of the two bodies attached by a joint. Absolute coordinates are employed by Orlandea *et al.* [7] and Avello *et al.* [11], and they are also used in the work of Shabana [12]. Examples of multibody codes which use relative coordinates include MESA-VERDE by Wittenburg [5, 6] and IMP by Uicker (see [10]). Relative coordinates tend to result in a smaller set of equations of motion.

A further important aspect of multibody codes is the method by which they describe how the bodies in a multibody system are linked together, that is, the method they use to describe the topology of a multibody system. For multibody codes which use absolute coordinates, all the necessary topological information is contained within these coordinates. Therefore, there is no need for a separate topological description of the system under study. However, for multibody codes which use relative coordinates there is a need for what McPhee [13] calls 'topological processing'.

The most popular method of performing this topological processing is to use graph theory [14]. Having said that, the extent to which graph theory is used in a multibody code varies significantly. For example, Wittenburg [5] uses graph theory purely to describe the topology of a multibody system, whereas Andrews has initiated a technique which uses graph theory to a much greater degree (see Andrews *et al.* [3] and Richard *et al.* [4]). This technique is called the 'vector-network method' and it employs a 'graph' of the multibody system which contains not only kinematic/topological information but also dynamic information. The equations of motion are then obtained directly from equations associated with this graph. Shi and McPhee [15] have continued the development of the vector-network method and have extended it to handle the case of flexible bodies by means of the bespoke DynaFlex code.

Over the last decade, the main area of development in multibody codes has been their extension to systems made up of flexible bodies. Multibody algorithms designed to deal with flexible bodies include those by Fisette *et al.* [16], Melzer [17] and Forehand *et al.* [18].

The aim of this paper is to describe and apply a notable existing multibody code. The code in question was developed by Roberson and Schwertassek and it is

described in detail in their book [19], which is considered to be a classic text within multibody circles. However, there are problems with the book: it is rather lengthy and involved, and it does not contain any *fully* worked examples of the multibody code applied to real-world problems. Indeed, fully worked examples of multibody codes applied to practical physical problems are rare in the literature. Consequently, this paper seeks to overcome these difficulties by providing an in-depth but concise overview of Roberson and Schwertassek's classical algorithm, together with a fully worked example of their multibody code applied to the double pendulum problem. That is, this paper allows the reader *quickly* to acquire a good understanding of Roberson and Schwertassek's formalism, which is not altogether possible by reading their excellent but long book [19].

Of course, the double pendulum is not a particularly difficult problem and its equations of motion can be obtained more simply using other methods, such as Lagrange's equations. However, it is informative to see how these equations of motion are derived from Roberson and Schwertassek's method and most of the complexity involved in this derivation can be attributed to the generality of the algorithm. Furthermore, this problem corresponds to applying what is essentially a three-dimensional code to a planar system.

The Roberson and Schwertassek method is a Eulerian method which is applicable to multibody systems made up of rigid and gyrostatic bodies. A gyrostatic body, or gyrostat for short, is a rigid body with an internal source of angular momentum. In order to describe these multibody systems, Roberson and Schwertassek's formalism uses relative coordinates instead of absolute coordinates.

Furthermore, for a certain class of multibody system, it will be shown later that Roberson and Schwertassek's algorithm *automatically* selects a set of governing, independent, generalised relative velocity coordinates and it also *automatically* produces a minimal set of purely differential equations of motion (i.e. the state-space equations of motion). These are a very desirable features when it is required to proceed to analytical solutions.

The Roberson and Schwertassek multibody method is discussed next in terms of how it may be usefully implemented. In this study the notation of Roberson and Schwertassek is adopted and whenever an equation comes directly from their study [19] its equation number there is displayed alongside its equation number in this paper.

Graph theory

In order to describe clearly and unambiguously the interconnection of the bodies in a multibody system, the Roberson and Schwertassek algorithm uses graph theory. A starting point for the algorithm is a drawing of the multibody system (see Fig. 1). Associated with this drawing is the corresponding system graph (see Fig. 2).

A graph consists of *vertices* linked by *edges*. The geometry of the edges within a graph is of no consequence. To obtain the system graph, each reference frame (body), including the inertial reference frame, is mapped to a vertex, and each joint is mapped to an edge. The edges in the graph are assigned arbitrary directions, as

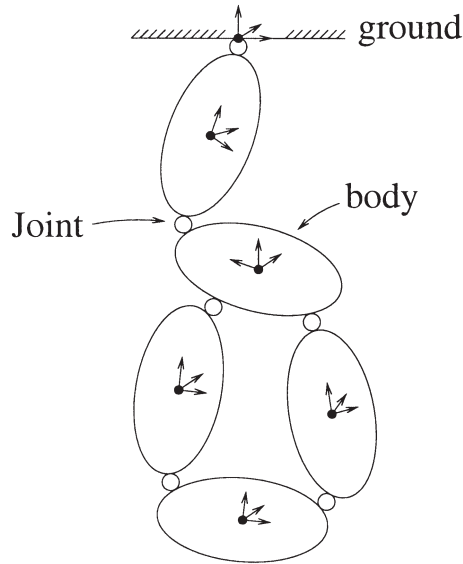


Fig. 1 An example of a multibody system with the body-fixed coordinate reference frames shown. The origin of each body-fixed reference frame is the centre of mass of the body.

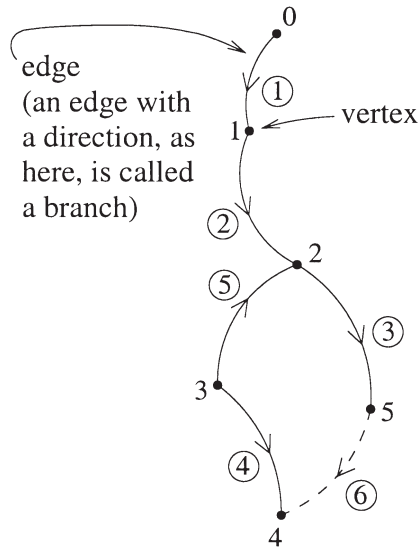


Fig. 2 The associated system graph for the system shown in Fig. 1. The edge (branch) labels are circled to distinguish them from the vertex labels.

shown in Fig. 2. An edge with a direction is called a *branch*. The vertex in the system graph which corresponds to the inertial frame is called the *reference* vertex and is assigned the number 0. Then the rest of the vertices in the graph are enumerated in an arbitrary way, as are the branches (see Fig. 2).

An important notion in graph theory is that of a *spanning tree*. A spanning tree is a connected subgraph of the graph which contains all the vertices of the original graph but has no closed loops (circuits). In Fig. 2, branches 3, 4, 5 and 6 constitute a circuit. By eliminating branch 6 from the graph a spanning tree is obtained. The branches of the original graph which are in the spanning tree are called the *arcs* and the branches of the original graph which are not in the spanning tree are called the *chords*. Therefore, in Fig. 2, branches 1, 2, 3, 4 and 5 are arcs and branch 6 is a chord. This is why, in Fig. 2, branch 6 is shown as dashed. Note, however, that the choice of spanning tree is not unique. In the example, branch 5 could have been eliminated from the original graph to obtain a spanning tree. In that case, branches 1, 2, 3, 4 and 6 would be the arcs and branch 5 would be the chord. Alternatively, either branch 3 or branch 4 could be eliminated from the original graph to obtain two more distinct spanning trees.

Having selected a spanning tree (for example, the spanning tree obtained by eliminating branch 6 in the above example) it is then possible to label the joints in the original multibody system. The joints associated with the arcs are called the *primary* joints and the joints associated with the chords are called the *secondary* joints.

Following the above, there is an important distinction in multibody dynamics between those systems which have closed loops (circuits), often referred to as closed *kinematic* loops by Roberson and Schwertassek, and those which do not. Systems with no closed kinematic loops are said to have a tree structure.

Now, let the set of all vertices, including the reference vertex (also called the 0-vertex), be denoted by \mathcal{O} and let the set of all vertices, not including the reference vertex, be \mathcal{N} . Furthermore, denote the set of branches, the set of arcs, and the set of chords by \mathcal{B} , \mathcal{A} and \mathcal{C} , respectively. In the system graph, let the number of vertices, besides the 0-vertex, be N_V and the number of branches be N_B . Thus, the vertices are numbered $0, 1, \dots, N_V$ and the branches are numbered $1, \dots, N_B$. Also, let the number of arcs be N_A and the number of chords be N_C . Then $N_B = N_A + N_C$.

In the multibody system, the number of bodies is denoted by N . Then N is equal to N_V . Also, since the number of branches in the spanning tree (the number of arcs) is one less than the number of vertices, N_A is equal to N_V . Collecting the above information together gives:

$$N = N_V = N_A \quad (1)$$

In particular, the number of primary joints (which is equal to the number of arcs) is equal to the number of bodies, N .

Matrices associated with the system graph

There are several matrices associated with the system graph which are used in Roberson and Schwertassek's multibody method. Two of the most important matrices are the incidence matrix, \bar{S} , and the path matrix, T . The incidence matrix

$$\bar{S} = [\bar{S}^{is}] \quad i \in \mathcal{O}, \quad s \in \mathcal{B} \quad (8.2.2-1a) \quad (2)$$

is of order $(N_V + 1) \times N_B$ and is defined as follows:

$$\bar{S}^{is} = \begin{cases} +1 & \text{if branch } s \text{ emanates from vertex } i \\ -1 & \text{if branch } s \text{ terminates on vertex } i \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The path matrix

$$T = [T^{ai}] \quad a \in \mathcal{A}, \quad i \in \mathcal{N} \quad (8.2.2-2) \quad (4)$$

is of order $N_A \times N_V$, hence square, and is defined as follows:

$$T^{ai} = \begin{cases} +1 & \text{if arc } a \text{ is on the direct chain from vertex } i \text{ to vertex } 0 \\ & \text{and arc } a \text{ is directed towards vertex } 0 \\ -1 & \text{if arc } a \text{ is on the direct chain from vertex } i \text{ to vertex } 0 \\ & \text{and arc } a \text{ is directed away from vertex } 0 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The direct chain from vertex i to vertex 0 is the sequence of arcs which link these two vertices.

For the above example, the incidence and path matrices are as follows:

$$\bar{S} = \begin{array}{c} \begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5 & 6 \\ \begin{array}{l} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \end{array} \end{array} \quad (6)$$

and

$$T = \begin{array}{c} \begin{array}{ccccc} & 1 & 2 & 3 & 4 & 5 \\ \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ 0 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array} \end{array} \quad (7)$$

Equations 6 and 7 are constructed by applying equations 3 and 5, respectively.

In equation 6, the numbers above \bar{S} are the branch labels and the numbers to the left of \bar{S} are the vertex labels. Similarly, in equation 7, the numbers above T are the vertex labels and the numbers to the left of T are the arc labels.

From equation 6, it can be seen that in each column of an incidence matrix \bar{S} there is one +1 entry, one -1 entry and the rest of the entries are zeros. Therefore, the sum of all the rows is a row of zeros, which means that \bar{S} does not have full rank.

It is, however, possible to create a matrix from \bar{S} which does have full rank. If the row of \bar{S} associated with the reference vertex (i.e. the 0 row, which is denoted S^{0*}) is removed, the following matrix is obtained:

$$S = [S^{is}] \quad i \in \mathcal{N}, \quad s \in \mathcal{B} \quad (8.2.2-1b)$$

This matrix is of order $N_V \times N_B$ and is called the *reduced* incidence matrix. The relationship between the full incidence matrix and the reduced incidence matrix is described by the following equation:

$$\bar{S} = \begin{bmatrix} S^{0*} \\ S \end{bmatrix} \quad (8.2.2-1c)$$

For the example above, the reduced incidence matrix is:

$$S = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \end{matrix} \quad (10)$$

It can be shown (see [19]) that the reduced incidence matrix, S , has full rank. That is, the rank of S is N_V .

From now on it will be assumed that the chords always receive higher branch labels than the arcs. This is the case for the example above, where the spanning tree was obtained by eliminating branch 6 (see Fig. 2). This assumption is not necessary for the algorithm, but it makes the rest of the analysis in this section more tractable.

Following the partitioning of the matrix \bar{S} into the matrices S^{0*} and S , the reduced incidence matrix, S , itself can be partitioned as:

$$S = [S_A \quad S_C] \quad (8.2.2-5c)$$

The columns of S_A are the columns of S which correspond to the arcs, $a \in \mathcal{A}$, and the columns of S_C are the columns of S which correspond to the chords, $c \in \mathcal{C}$. That is:

$$S_A = [S^{ia}] \quad i \in \mathcal{N}, \quad a \in \mathcal{A} \quad (8.2.2-5a)$$

and

$$S_C = [S^{ic}] \quad i \in \mathcal{N}, \quad c \in \mathcal{C} \quad (8.2.2-5b)$$

The matrix S_A is of order $N_V \times N_A$ and hence, from equation 1, it is square. It can be shown (see [19]) that T is the inverse of S_A . This means that there is no need to calculate T directly from first principles using equation 5. Instead, the full incidence

matrix, \bar{S} , can be constructed using equation 3. This matrix can then be partitioned to obtain the reduced incidence matrix, S , which in turn can be partitioned to give S_A . Finally, on inverting S_A , the path matrix, T , can be obtained. The reason for calculating T from \bar{S} in this way is because direct evaluation of T from equation 5 is not simple to automate.

The unreduced dynamic equations and relative coordinates

The Roberson and Schwertassek algorithm produces two sets of differential equations which govern the motion of the multibody system. The first set are the *kinematic* equations of motion. These are first-order differential equations which naturally relate the system position and velocity variables. The second set of equations are the *dynamic* equations of motion. These are first-order differential equations in the velocity variables. In this paper, only the dynamic equations of motion will be considered, because for the example problem given here the kinematic equations of motion are trivial.

As stated earlier, the multibody system under study consists of N bodies. The motion of each one of these bodies can be described using six *absolute* velocity coordinates: three translational and three rotational. Consequently, $6N$ absolute velocity coordinates can be used to describe the motion of the multibody system. For these $6N$ velocity coordinates, $6N$ dynamic equations of motion can be written down quite easily. However, not all of these velocity coordinates are independent, because the joints in the multibody system will necessarily impose constraints on the relative motion of the bodies. The presence of such joints also means that the $6N$ dynamic equations of motion contain unknown constraint forces, making these them difficult to solve.

In the previous section, it was shown that once a spanning tree has been selected there are $N_A = N$ primary joints. The idea behind Roberson and Schwertassek's method is to take the above *unreduced* dynamic equations in terms of the absolute velocity coordinates and then reformulate them in terms of relative velocity coordinates across the primary joints. The term 'unreduced' refers to the fact that these equations are in terms of coordinates which are not all independent.

Across each primary joint (arc) a , six relative velocity coordinates can be used. In Roberson and Schwertassek's method, the following six relative velocity coordinates are employed:

$$G^a = \begin{bmatrix} \Omega^a \\ V^a \end{bmatrix} \quad (11.1.6-12a) \quad (14)$$

where G^a is a 6×1 matrix and where Ω^a and V^a are 3×1 matrices. The quantities Ω^a and V^a are the components of the relative angular velocity and the relative translational velocity across joint a , respectively. These components are, with respect to the frame, fixed in the body (vertex) from which primary joint (arc) a emanates. Since there are N primary joints, $6N$ such relative velocity coordinates can be used to describe the motion of the multibody system. Consequently, the $6N$ unreduced

dynamic equations in terms of the $6N$ *absolute* velocity coordinates are reformulated into $6N$ unreduced dynamic equations in terms of $6N$ *relative* velocity coordinates. These reformulated equations are:

$$\sum_{b \in \mathcal{A}} J^{ab} \dot{G}^b = Q^a + \Lambda^a + \sum_{c \in \mathcal{C}} W^{ac} \Lambda^c \quad \text{for } a \in \mathcal{A} \quad (11.1.7-2) \tag{15}$$

In equation 15, the superscripts a and b are indices running over the arcs of the system graph (primary joints of the multibody system), and superscript c is an index running over the chords of the system graph (secondary joints of the multibody system).

On the left-hand side (LHS) of equation 15, J^{ab} is a 6×6 generalised inertia matrix and \dot{G}^b is the time derivative (note, in this study, dot denotes time derivative) of the 6×1 matrix containing the relative velocity coordinates across arc b (i.e. equation 14 with a replaced by b).

On the right-hand side (RHS) of equation 15, Λ^a is a 6×1 matrix which contains the generalised *interaction* forces and torques across primary joint (arc) a . Included in these actions are the unknown constraint forces and torques across the joint. Also, Q^a is a 6×1 matrix which contains the generalised *externally applied* forces and torques, as well as all the remaining actions, such as the centripetal and Coriolis terms. Finally, on the RHS of equation 15, the sum is related to the *kinematic consistency conditions*. These conditions ensure that the circuits in the multibody system are indeed *closed* loops. The quantity Λ^c is a 6×1 matrix which contains the generalised interaction forces and torques across secondary joint (chord) c , and the quantity W^{ac} is a 6×6 matrix whose physical significance will become apparent from the next paragraph.

In order to complete the description of the motion of the multibody system, the aforementioned kinematic consistency conditions must be adjoined to the unreduced dynamic equations of motion given in equation 15. These kinematic consistency conditions are algebraic constraint equations which express the relative velocities across the secondary joints (chords) in terms of the relative velocities across the primary joints (arcs), and they take the form:

$$G^c = \sum_{a \in \mathcal{A}} W^{acT} G^a \quad \text{for } c \in \mathcal{C} \quad (11.1.7-3) \tag{16}$$

where the superscript T denotes transpose. That is, W^{acT} is the transpose of the 6×6 matrix W^{ac} . The LHS of equation 16 is obtained from equation 14 by replacing a with c .

Although the underlying ideas behind equations 15 and 16 are quite simple, their derivation is algebraically very complicated and consequently it is not reproduced here. The interested, and brave, reader is referred to Roberson and Schwertassek's book [19] for this derivation.

In the present study, Roberson and Schwertassek's method is applied to the well known double pendulum problem for purposes of illustration. This problem has a tree structure and so only systems with no closed kinematic loops will be consid-

ered further. As a result of this, the system graph will have no chords, which in turn means that equation 16 disappears entirely and equation 15 simplifies to:

$$\sum_{b \in \mathcal{A}} J^{ab} \dot{G}^b = Q^a + \Lambda^a \quad \text{for } a \in \mathcal{A} \quad (11.3.1-1b) \quad (17)$$

The next step is to reduce the unreduced dynamic equations (shown compactly in equation 17) to state–space form. That is, the unknown constraint forces are eliminated from these equations by reformulating them in terms of a set of *independent* generalised relative velocity coordinates. At the moment, equation 17 is expressed in terms of the $6N$ relative velocity coordinates and these are a set of *dependent* generalised velocity coordinates.

Equation 17 is very well suited to this elimination process. Firstly, the equation is expressed in terms of relative velocity coordinates, as opposed to absolute velocity coordinates, and, as explained later, this makes it easier to eliminate the unknown constraint forces. Secondly, for each primary joint (arc) a , equation 17 has been especially formulated so that only the interaction forces and torques across joint a (which contain the unknown constraint forces across joint a) appear in the equation. That is, only the Λ corresponding to joint a , namely Λ^a , appears in equation 17.

In the next section it will be shown how the unknown constraint forces are eliminated from equation 17 so that it can be reduced to state–space form.

Free mode vectors and the reduction to state–space form

In order to reduce the unreduced dynamic equations of motion to state–space form, Roberson and Schwertassek used what they termed the *free mode vectors*. Each kind of joint has its associated free mode vectors and these are used to eliminate the unknown constraint forces and torques acting across a joint of that type.

A good way to illustrate free mode vectors is through a simple example. The example chosen here is that of a revolute joint rotating about an axis parallel to the z axis of the body from which the joint emanates (see Figs 3, 4 and 5).

Let the joint be called joint a and let the bodies at the ‘beginning’ and ‘end’ of the joint be bodies 1 and 2, respectively. To make the problem a little more complicated, it is assumed that joint a is also subject to a viscous damping moment, $-d\dot{\theta}$ (where d is a positive constant), as shown in Fig. 5. Therefore, the interaction forces and torques acting across joint a on body 2 consist of the *unknown* constraint forces in the x and y directions, namely F_x and F_y , and the viscous damping moment (which is a *known* function of $\dot{\theta}$).

The free mode vectors relate the *independent* generalised relative velocity coordinates across the joint to the *dependent* generalised relative velocity coordinates across the joint. That is, if g^a is a column vector containing the independent generalised relative velocity coordinates across joint a and ϕ^a is a matrix whose columns are the free mode vectors, then the dependent generalised relative velocity coordinates across joint a , G^a , are given by:

$$G^a = \phi^a g^a \quad (18)$$

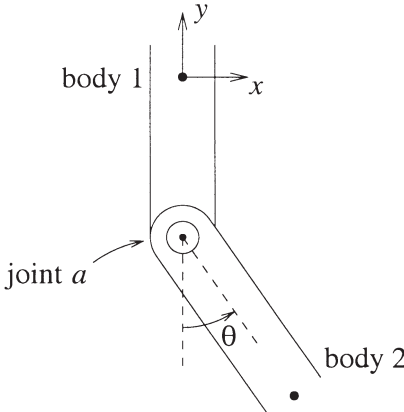


Fig. 3 Joint a (a revolute joint).

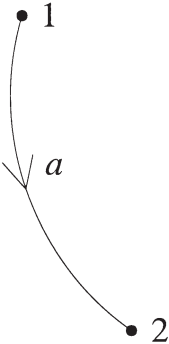


Fig. 4 Associated system graph for the system shown in Fig. 3.

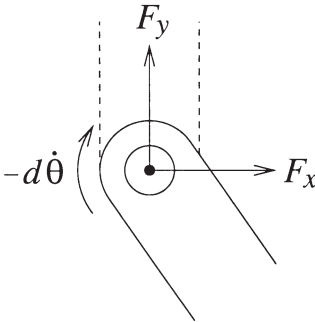


Fig. 5 Interaction forces and torques acting across joint a on body 2 from Fig. 3.

Equation 18 is a simplified version of equation (11.2.1–2) in [19]. In this particular case, there is only one independent generalised relative velocity coordinate across joint a , namely $\dot{\theta}$, and so:

$$g^a = \dot{\theta} \quad (19)$$

Furthermore, using equation 14, the dependent generalised relative velocity coordinates across the joint are as follows:

$$G^a = \begin{bmatrix} \Omega^a \\ V^a \end{bmatrix} = [0 \ 0 \ \dot{\theta} \ 0 \ 0 \ 0]^T \quad (20)$$

The reason for the first two zeros on the RHS of equation 20 is that the pivot does not rotate about the x or y axes, and the reason for the last three zeros is that there is no translational velocity across the pivot. Expressing this in another way, these zeros correspond to the ‘locked’ modes of the joint.

Finally, by substituting equations 19 and 20 into equation 18, the free mode vector for the revolute joint shown in Fig. 3 is obtained:

$$\phi^a = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T \quad (21)$$

This free mode vector can then be used to eliminate the unknown constraint forces.

As stated earlier, the interaction forces and torques acting across joint a are contained in the 6×1 matrix Λ^a . To be more specific, Λ^a is given by the formula:

$$\Lambda^a = \begin{bmatrix} L^a \\ F^a \end{bmatrix} \quad (9.2.3-21a) \quad (22)$$

where L^a and F^a are 3×1 matrices which contain the components of the interaction torque and interaction force across joint a , respectively. These components are, with respect to the frame, fixed in the body from which joint a emanates. For the example shown in Fig. 5, equation 22 becomes:

$$\Lambda^a = [0 \ 0 \ -d\dot{\theta} \ F_x \ F_y \ 0]^T \quad (23)$$

and on premultiplying this by ϕ^{aT} , where ϕ^a comes from equation 21, the following equation is obtained:

$$\phi^{aT} \Lambda^a = -d\dot{\theta} \quad (24)$$

Thus the *unknown* constraint forces have been eliminated but the *known* viscous damping moment is retained.

The procedure illustrated in the above example can also be applied to the *general* unreduced dynamic equations of motion for a system with a tree structure (see equation 17) in order to reduce them to state–space form and eliminate the unknown constraint forces and torques. This process is described in the following.

Firstly, both sides of equation 17 are premultiplied by ϕ^{aT} , giving:

$$\sum_{b \in \mathcal{A}} \phi^{aT} J^{ab} \dot{G}^b = \phi^{aT} Q^a + \phi^{aT} \Lambda^a \quad \text{for } a \in \mathcal{A} \quad (25)$$

Then the time derivative of equation 18, with a replaced by b , is substituted into equation 25 to obtain:

$$\sum_{b \in \mathcal{A}} \phi^{aT} J^{ab} (\dot{\phi}^b \dot{g}^b + \dot{\phi}^b g^b) = \phi^{aT} Q^a + \phi^{aT} \Lambda^a \quad \text{for } a \in \mathcal{A} \quad (26)$$

Finally, the above equation is rearranged to become:

$$\sum_{b \in \mathcal{A}} \phi^{aT} J^{ab} \dot{\phi}^b \dot{g}^b = \phi^{aT} \left(Q^a - \sum_{b \in \mathcal{A}} J^{ab} \dot{\phi}^b g^b \right) + \phi^{aT} \Lambda^a \quad \text{for } a \in \mathcal{A} \quad (27)$$

Thus the dynamic *state-space* equations of motion for a multibody system with a tree structure are:

$$\sum_{b \in \mathcal{A}} J_I^{ab} \dot{g}^b = \chi^a + \lambda^a \quad \text{for } a \in \mathcal{A} \quad (11.2.2-1a) \quad (28)$$

where

$$J_I^{ab} = \phi^{aT} J^{ab} \phi^b \quad (11.2.2-2a) \quad (29)$$

$$\chi^a = \phi^{aT} X^a \quad \text{and} \quad X^a = Q^a - \sum_{b \in \mathcal{A}} J^{ab} \dot{\phi}^b g^b \quad (30)$$

and where

$$\lambda^a = \phi^{aT} \Lambda^a \quad (9.2.3-21c) \quad (31)$$

Equation 30 comes from equations 11.2.2-2b and 11.2.1-8 in [19]. In equation 31, as in equation 24, the unknown constraint forces in Λ^a have been eliminated by ϕ^a and do not appear in λ^a .

At this point it is useful to make some observations about the above process of reducing the unreduced dynamic equations of motion to state-space form.

The first observation is that, for multibody systems with tree structures, Roberson and Schwertassek’s algorithm automatically selects a set of governing, independent, generalised relative velocity coordinates, namely:

$$g^a \quad \text{for } a \in \mathcal{A} \quad (32)$$

This is a strong feature of the method, and not present in any other algorithms, which leave the problem of selecting a set of independent generalised coordinates describing the system to the user, with all the scope for human error that that entails.

Secondly, in the previous section it was mentioned that it is easier to reduce the unreduced dynamic equations of motion to state-space form when they are expressed in terms of relative rather than absolute coordinates. The reason for this is that the free mode vectors are very much simpler when relative, as opposed to absolute, coordinates are used problem. This simplicity is evident in the free mode vector for the revolute joint shown in Fig. 3 (see equation 21).

Thirdly, the reduction to state-space form for systems with closed kinematic loops is a much more difficult process than for systems with tree structures. Indeed, some

three-dimensional systems with closed kinematic loops cannot be reduced to state–space form at all. The reduction process can start the same way as it was started for systems with a tree structure, that is, by premultiplying both sides of equation 15 by ϕ^{aT} . However, this will eliminate only the unknown constraint forces and torques acting across primary joint a , contained in Λ^a , and it will not eliminate the unknown constraint forces and torques acting across each secondary joint c , contained in Λ^c . Consequently, for systems with closed kinematic loops, it may be necessary to leave the equations of motion in differential algebraic form.

The double pendulum problem

On the basis of the foregoing discussion it is now possible to apply Roberson and Schwertassek's algorithm. In this study the method is applied to the well known double pendulum problem. This problem has a tree structure and is shown in Figs 6–9. By definition, in multibody systems which have a tree structure, all the branches of the system graph are arcs and therefore all the joints are primary joints.

It should be pointed out that the free body diagrams of the two bars which make up the double pendulum, shown in Figs 8 and 9, are not required by the algorithm. They are presented here merely to show the *known* externally applied forces and the *unknown* constraint forces acting on the bars.

In the rest of this paper, unless otherwise stated, equations which do not have an equation number from Roberson and Schwertassek's study [19] are not general equations but are specific to the double pendulum problem.

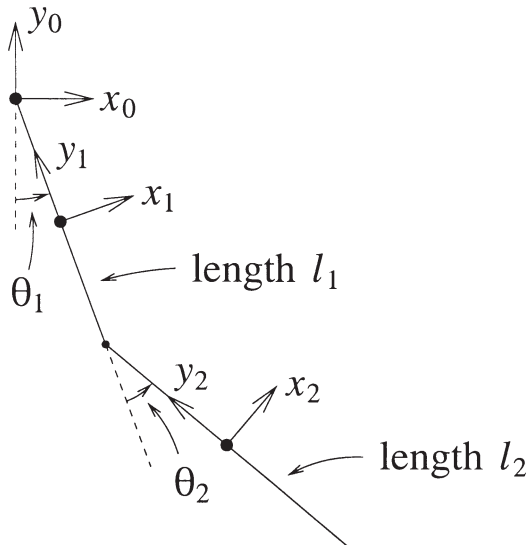


Fig. 6 Schematic of a double pendulum, showing the inertial reference frame and the two body-fixed frames.

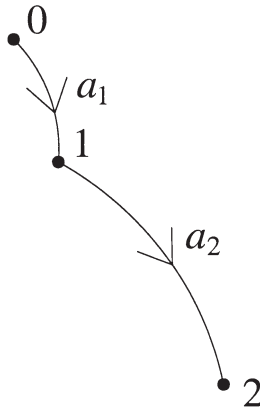


Fig. 7 The associated system graph for the double pendulum. Arc a_1 represents the top pivot (joint), arc a_2 represents the bottom pivot (joint), and vertices 0, 1 and 2 represent the three frames (bodies), labelled 0, 1 and 2.

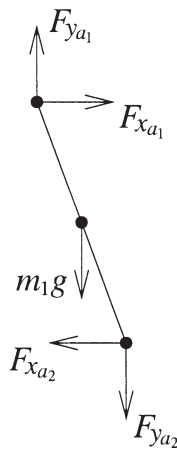


Fig. 8 Free body diagram of the upper bar (bar 1) of the double pendulum.

The double pendulum consists of two uniform thin bars of mass m_1 and m_2 , and lengths l_1 and l_2 , respectively. As can be seen from Fig. 7, the primary joints (arcs) are labelled a_1 and a_2 , instead of using the numbers 1 and 2. The reason for this is to distinguish between the arc labelling and the vertex labelling. In the present case, the elements of the reduced incidence matrix and the path matrix are as follows:

$$\begin{aligned} \bar{S}^{1a_1} &= -1 & \bar{S}^{1a_2} &= 1 \\ \bar{S}^{2a_1} &= 0 & \bar{S}^{2a_2} &= -1 \end{aligned} \tag{33}$$

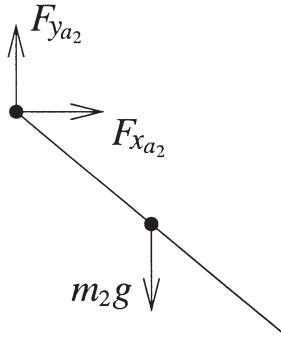


Fig. 9 Free body diagram of the lower bar (bar 2) of the double pendulum.

and

$$\begin{aligned}
 T^{a_1} &= -1 & T^{a_1 2} &= -1 \\
 T^{a_2 1} &= 0 & T^{a_2 2} &= -1
 \end{aligned}
 \tag{34}$$

where the above equations were constructed using equations 3 and 5.

In order to calculate the dynamic state–space equations of motion for the double pendulum it is necessary to evaluate equation 28 for this problem. To reiterate what was stated earlier in this paper, the superscripts *a* and *b*, which appear in equations such as equation 28, are indices running over the arcs of the system graph. That is $a, b \in \mathcal{A}$, and therefore for the double pendulum problem, *a* and *b* can take the values a_1 or a_2 . Also, unless otherwise stated, in the following subsections, the indices *i*, *j* and *k* run over the bodies of the multibody system. This means that i, j and $k \in \mathcal{N}$, and consequently in the present case *i*, *j* and *k* can take the values 1 or 2.

In the following subsections each term in equation 28 is considered in turn, with a view to evaluating it for the double pendulum problem.

The \dot{g}^b term

The quantity g^b is the independent generalised relative velocity coordinate across joint *b*, and *b* can take the values a_1 or a_2 . Consequently:

$$g^{a_1} = \dot{\theta}_1 \quad \text{and} \quad g^{a_2} = \dot{\theta}_2
 \tag{35}$$

The J_i^{ab} term

The J_i^{ab} term is a generalised inertia term, as defined in equation 29:

$$J_i^{ab} = \phi^{aT} J^{ab} \phi^b$$

where, as stated previously, J^{ab} is a 6×6 matrix. Since *a* and *b* can take the values a_1 or a_2 , the quantities $J_i^{a_1 a_1}$, $J_i^{a_1 a_2}$, $J_i^{a_2 a_1}$ and $J_i^{a_2 a_2}$ are being sought.

In this problem both joints are revolute joints which rotate about axes parallel to the z axis of the body from which that joint emanates. Therefore, from equation 21, the free mode vectors for joints a_1 and a_2 are:

$$\phi^{a_1} = \phi^{a_2} = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T \tag{36}$$

and consequently, for the present case, equation 29 becomes:

$$J_I^{ab} = \{J^{ab}\}_{33} \tag{37}$$

where $\{\cdot\}_{ij}$ means the i, j th element of the two-dimensional matrix inside the brackets.

It is now necessary to consider the definition of the matrix J^{ab} , which has not previously been dealt with in this paper. This 6×6 matrix is defined by the following equation:

$$J_I^{ab} = \begin{bmatrix} J_{11}^{ab} & J_{12}^{ab} \\ J_{21}^{ab} & J_{22}^{ab} \end{bmatrix} \tag{11.1.6-1a} \tag{38}$$

where $J_{11}^{ab}, J_{12}^{ab}, J_{21}^{ab}$ and J_{22}^{ab} are all 3×3 matrices. Therefore, combining the above equation with equation 37 yields for the double pendulum:

$$J_I^{ab} = \{J_{11}^{ab}\}_{33} \tag{39}$$

The next step is to consider the definition of the 3×3 matrix J_{11}^{ab} . This is given by:

$$J_{11}^{ab} = A^{f(a)0} \left(\sum_{i,k \in \mathcal{N}} T^{ai} T^{bk} K^{ik} \right) A^{0f(b)} \tag{11.1.5-15} \tag{40}$$

where A^{ij} is the 3×3 direction cosine matrix from the frame fixed in body j to the frame fixed in body i , and $f(a)$ is the body from which joint (arc) a originates. For this problem:

$$\begin{aligned} A^{f(a_1)0} &= A^{00} = E \\ A^{f(a_2)0} &= A^{10} = \begin{bmatrix} \cos \theta_1 & \sin \theta_1 & 0 \\ -\sin \theta_1 & \cos \theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ A^{0f(a_1)} &= A^{00} = E \\ A^{0f(a_2)} &= A^{01} = (A^{10})^T \end{aligned} \tag{41}$$

where E is the 3×3 identity or unity matrix.

Also is equation 40, K^{ik} is defined to be the following 3×3 matrix:

$$K^{ik} = A^{0i} I^i \delta^{ik} A^{i0} + \sum_{j \in \mathcal{N}} m_j \left((A^{0k} \hat{a}^{kj})^T (A^{0i} \hat{a}^{ij}) E - A^{0k} \hat{a}^{kj} \hat{a}^{ijT} A^{i0} \right) \tag{11.1.5-13a} \tag{42}$$

where I^i is the inertia matrix of body i about its center of mass (COM) and resolved in the frame fixed in that body. For this problem:

$$I^i = \begin{bmatrix} \frac{m_i l_i^2}{12} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{m_i l_i^2}{12} \end{bmatrix} \tag{43}$$

Furthermore, in equation 42, δ^{jk} is the Kronecker delta symbol and the quantities \hat{a}^{ij} are related to what Roberson and Schwertassek call the *attachment vectors*. These attachment vectors will be discussed later.

Since the only non-zero entry in the last row or the last column of the direction cosine matrices in equation 41 is the number 1 in the 3,3 position, substituting equation 40 into equation 39 results in:

$$J_i^{ab} = \left\{ \sum_{i,k \in \mathcal{N}} T^{ai} T^{bk} K^{ik} \right\}_{33} \tag{44}$$

Therefore, using equation 34 yields:

$$\begin{aligned} J_i^{a_1 a_1} &= \{ T^{a_1 1} T^{a_1 1} K^{11} + T^{a_1 1} T^{a_1 2} K^{12} + T^{a_1 2} T^{a_1 1} K^{21} + T^{a_1 2} T^{a_1 2} K^{22} \}_{33} \\ &= \{ K^{11} + K^{12} + K^{21} + K^{22} \}_{33} \end{aligned} \tag{45}$$

$$\begin{aligned} J_i^{a_1 a_2} &= \{ T^{a_1 1} T^{a_2 1} K^{11} + T^{a_1 1} T^{a_2 2} K^{12} + T^{a_1 2} T^{a_2 1} K^{21} + T^{a_1 2} T^{a_2 2} K^{22} \}_{33} \\ &= \{ K^{12} + K^{22} \}_{33} \end{aligned} \tag{46}$$

$$\begin{aligned} J_i^{a_2 a_1} &= \{ T^{a_2 1} T^{a_1 1} K^{11} + T^{a_2 1} T^{a_1 2} K^{12} + T^{a_2 2} T^{a_1 1} K^{21} + T^{a_2 2} T^{a_1 2} K^{22} \}_{33} \\ &= \{ K^{21} + K^{22} \}_{33} \end{aligned} \tag{47}$$

$$\begin{aligned} J_i^{a_2 a_2} &= \{ T^{a_2 1} T^{a_2 1} K^{11} + T^{a_2 1} T^{a_2 2} K^{12} + T^{a_2 2} T^{a_2 1} K^{21} + T^{a_2 2} T^{a_2 2} K^{22} \}_{33} \\ &= \{ K^{22} \}_{33} \end{aligned} \tag{48}$$

The consequence of the above equations is that in order to evaluate $J_i^{a_1 a_1}$, $J_i^{a_1 a_2}$, $J_i^{a_2 a_1}$ and $J_i^{a_2 a_2}$, it is necessary to calculate $\{K^{11}\}_{33}$, $\{K^{12}\}_{33}$, $\{K^{21}\}_{33}$ and $\{K^{22}\}_{33}$.

Returning to the attachment vectors, the vectors $\hat{\mathbf{a}}^{ij}$ are related to what Roberson and Schwertassek call the *modified attachment vectors*, $\hat{\mathbf{c}}^{ia}$, via the equation:

$$\hat{\mathbf{a}}^{ij} = \sum_{a \in \mathcal{A}} \bar{S}^{ia} T^{aj} \hat{\mathbf{c}}^{ia} \quad \text{for } i \in \mathcal{O} \text{ and } j \in \mathcal{N} \tag{49}$$

Equation 49 comes from equations 10.1.4-3a and 10.1.2-5 in [19]. The modified attachment vector, $\hat{\mathbf{c}}^{ia}$, is defined only if joint a is incident on body i . If this is the case, then $\hat{\mathbf{c}}^{ia}$ is the vector from the COM of body i to the attachment *point* of joint a on the body to which that joint *points* (see Figs 10 and 11).

The \hat{a}^{ij} in equation 42 are the vectors $\hat{\mathbf{a}}^{ij}$ resolved in the frame fixed in body i . This means that, from equation 49:

$$\hat{a}^{ij} = \sum_{a \in \mathcal{A}} \bar{S}^{ia} T^{aj} \hat{c}^{ia} \quad \text{for } i \in \mathcal{O} \text{ and } j \in \mathcal{N} \tag{50}$$

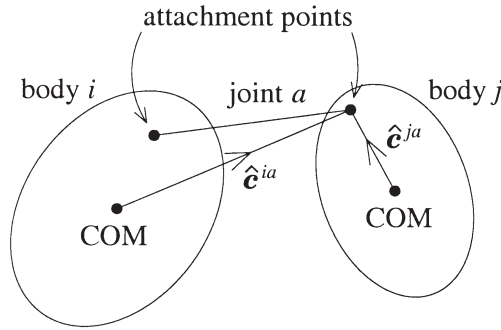


Fig. 10 Joint a , which links body i to body j , and the modified attachment vectors, \hat{c}^{ia} and \hat{c}^{ja} .

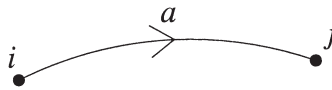


Fig. 11 The associated system graph for the system shown in Fig. 10. Note that here the joint points to body j .

where \hat{c}^{ja} are also \hat{c}^{ia} resolved in the frame fixed in body i . For this problem:

$$\hat{c}^{1a_1} = [0 \quad l_1/2 \quad 0]^T \quad \hat{c}^{1a_2} = [0 \quad -l_1/2 \quad 0]^T$$

$$\hat{c}^{2a_1} \leftarrow \text{not defined because joint } a_1 \quad \hat{c}^{2a_2} = [0 \quad l_2/2 \quad 0]^T$$

$$\text{is not incident on body 2} \tag{51}$$

Therefore, using equations 33, 34 and 51 in equation 50 gives:

$$\hat{a}^{11} = \bar{S}^{1a_1} T^{a_1 1} \hat{c}^{1a_1} + \bar{S}^{1a_2} T^{a_2 1} \hat{c}^{1a_2} = \hat{c}^{1a_1} = [0 \quad l_1/2 \quad 0]^T \tag{52}$$

$$\hat{a}^{12} = \bar{S}^{1a_1} T^{a_1 2} \hat{c}^{1a_1} + \bar{S}^{1a_2} T^{a_2 2} \hat{c}^{1a_2} = \hat{c}^{1a_1} - \hat{c}^{1a_2} = [0 \quad l_1 \quad 0]^T \tag{53}$$

$$\hat{a}^{21} = \bar{S}^{2a_1} T^{a_1 1} \hat{c}^{2a_1} + \bar{S}^{2a_2} T^{a_2 1} \hat{c}^{2a_2} = 0 \tag{54}$$

$$\hat{a}^{22} = \bar{S}^{2a_1} T^{a_1 2} \hat{c}^{2a_1} + \bar{S}^{2a_2} T^{a_2 2} \hat{c}^{2a_2} = \hat{c}^{2a_2} = [0 \quad l_2/2 \quad 0]^T \tag{55}$$

In order to proceed further, a similar argument is applied to equation 42 as was applied to equation 40 to obtain equation 44. This yields:

$$\{K^{ik}\}_{33} = \{I^i \delta^{ik}\}_{33} + \sum_{j \in \mathcal{N}} m_j \left((A^{0k} \hat{a}^{kj})^T (A^{0i} \hat{a}^{ij}) - \{\hat{a}^{kj} \hat{a}^{ijT}\}_{33} \right) \tag{56}$$

However, since the only non-zero elements of \hat{a}^{kj} and \hat{a}^{ij} are the middle elements (see equations 52–55) it is found that:

$$\{\hat{a}^{kj} \hat{a}^{ijT}\}_{33} = 0 \tag{57}$$

This then means that equation 56 becomes:

$$\{K^{ik}\}_{33} = \{I^i \delta^{ik}\}_{33} + \sum_{j \in \mathcal{N}} m_j (A^{0k} \hat{a}^{kj})^T (A^{0i} \hat{a}^{ij}) \quad (58)$$

Furthermore, by examining the inertia matrix in equation 43 for each of the bars, it is seen that:

$$\{I^1\}_{33} = \frac{m_1 l_1^2}{12} \quad \text{and} \quad \{I^2\}_{33} = \frac{m_2 l_2^2}{12} \quad (59)$$

Consequently, from equation 58 the following equations result:

$$\{K^{11}\}_{33} = \frac{m_1 l_1^2}{12} + m_1 (A^{01} \hat{a}^{11})^T (A^{01} \hat{a}^{11}) + m_2 (A^{01} \hat{a}^{12})^T (A^{01} \hat{a}^{12}) \quad (60)$$

$$\{K^{12}\}_{33} = m_1 (A^{02} \hat{a}^{21})^T (A^{01} \hat{a}^{11}) + m_2 (A^{02} \hat{a}^{22})^T (A^{01} \hat{a}^{12}) \quad (61)$$

$$\{K^{21}\}_{33} = m_1 (A^{01} \hat{a}^{11})^T (A^{02} \hat{a}^{21}) + m_2 (A^{01} \hat{a}^{12})^T (A^{02} \hat{a}^{22}) \quad (62)$$

$$\{K^{22}\}_{33} = \frac{m_2 l_2^2}{12} + m_1 (A^{02} \hat{a}^{21})^T (A^{02} \hat{a}^{21}) + m_2 (A^{02} \hat{a}^{22})^T (A^{02} \hat{a}^{22}) \quad (63)$$

By using equations 41 and 52–55, together with the fact that A^{02} is A^{01} with θ_1 replaced by $\theta_1 + \theta_2$, it can be shown that:

$$A^{01} \hat{a}^{11} = \frac{l_1}{2} [-\sin \theta_1 \quad \cos \theta_1 \quad 0]^T \quad (64)$$

$$A^{01} \hat{a}^{12} = l_1 [-\sin \theta_1 \quad \cos \theta_1 \quad 0]^T \quad (65)$$

$$A^{02} \hat{a}^{21} = 0 \quad (66)$$

$$A^{02} \hat{a}^{22} = \frac{l_2}{2} [-\sin(\theta_1 + \theta_2) \quad \cos(\theta_1 + \theta_2) \quad 0]^T \quad (67)$$

Substituting the above equations into equations 60–63 then leads to:

$$\{K^{11}\}_{33} = \frac{m_1 l_1^2}{12} + \frac{m_1 l_1^2}{4} + m_2 l_1^2 = \left(\frac{m_1}{3} + m_2 \right) l_1^2 \quad (68)$$

$$\{K^{12}\}_{33} = \frac{m_2 l_1 l_2}{2} (\sin \theta_1 \sin(\theta_1 + \theta_2) + \cos \theta_1 \cos(\theta_1 + \theta_2)) = \frac{m_2 l_1 l_2}{2} \cos \theta_2 \quad (69)$$

$$\{K^{21}\}_{33} = \frac{m_2 l_1 l_2}{2} \cos \theta_2 \quad (70)$$

$$\{K^{22}\}_{33} = \frac{m_2 l_2^2}{12} + \frac{m_2 l_2^2}{4} = \frac{m_2 l_2^2}{3} \quad (71)$$

Finally, the generalised inertia terms can be obtained by substituting equations 68–71 into equations 45–48 to give:

$$J_I^{a_1 a_1} = \left(\frac{m_1}{3} + m_2 \right) l_1^2 + m_2 l_1 l_2 \cos \theta_2 + \frac{m_2 l_2^2}{3} \tag{72}$$

$$J_I^{a_1 a_2} = \frac{m_2 l_1 l_2}{2} \cos \theta_2 + \frac{m_2 l_2^2}{3} \tag{73}$$

$$J_I^{a_2 a_1} = \frac{m_2 l_1 l_2}{2} \cos \theta_2 + \frac{m_2 l_2^2}{3} \tag{74}$$

$$J_I^{a_2 a_2} = \frac{m_2 l_2^2}{3} \tag{75}$$

The χ^a term

The next step is to consider the first term on the RHS of equation 28. This term is the χ^a term and because a can take the values a_1 or a_2 the quantities χ^{a_1} and χ^{a_2} are sought. The first part of equation 30 stated that:

$$\chi^a = \phi^{aT} X^a$$

and the second part that the 6×1 matrix X^a is given by:

$$X^a = Q^a - \sum_{b \in \mathcal{A}} J^{ab} \dot{\phi}^b g^b$$

For the double pendulum problem, the sum on the RHS of the above equation is zero because the free mode vectors are constant (see equation 36). Therefore, the first part of equation 30 becomes:

$$\chi^a = \phi^{aT} Q^a \tag{76}$$

The form of the free mode vectors in equation 36 also means that for this problem equation 76 reduces to:

$$\chi^a = \{Q^a\}_3 \tag{77}$$

where $\{\cdot\}_i$ denotes the i th element of the column matrix inside the brackets. The 6×1 column matrix Q^a is defined by the following equation:

$$Q^a = \begin{bmatrix} Q_1^a \\ Q_2^a \end{bmatrix} \tag{11.1.7-1b} \tag{78}$$

where Q_1^a and Q_2^a are 3×1 column matrices. Consequently, using equations 77 and 78 results in:

$$\chi^a = \{Q_1^a\}_3 \tag{79}$$

In order to proceed further, it is necessary to look at the definition of Q_1^a . This matrix, together with the matrix Q_2^a , is defined by the equation:

$$Q_r^a = A^{f(a)0} \sum_{i \in \mathcal{N}} T^{ai} (Q_r''^i + Q_r'^i) \quad \text{for } r = 1, 2 \tag{11.1.5-27} \tag{80}$$

where $Q_r''^i$ and $Q_r'^i$ are 3×1 column matrices (note that the prime and double prime do not refer to differentiation). Then, by using a similar argument to that used when equation 44 was obtained from equations 39 and 40, equation 80 can be substituted into equation 79 to give:

$$\chi^a = \left\{ \sum_{i \in \mathcal{N}} T^{ai} (Q_r''^i + Q_r'^i) \right\}_3 \tag{81}$$

The above equation can be further simplified for the double pendulum problem by employing equation 34, with the result that:

$$\chi^{a1} = \{T^{a11}(Q_1''^1 + Q_1'^1) + T^{a12}(Q_1''^2 + Q_1'^2)\}_3 = -\{Q_1''^1 + Q_1'^1 + Q_1''^2 + Q_1'^2\}_3 \tag{82}$$

and

$$\chi^{a2} = \{T^{a21}(Q_1''^1 + Q_1'^1) + T^{a22}(Q_1''^2 + Q_1'^2)\}_3 = -\{Q_1''^2 + Q_1'^2\}_3 \tag{83}$$

Consequently, in order to calculate χ^{a1} and χ^{a2} it is necessary to calculate $Q_1''^1$, $Q_1''^2$, $Q_1'^1$ and $Q_1'^2$.

Calculating the $Q_1''^1$ and $Q_1''^2$ terms

Consider first the terms $Q_1''^i$ (for $i = 1, 2$), which are given by the equation:

$$Q_1''^i = Q_{II}^i - \sum_{j \in \mathcal{N}} K^{ij} Q_{III}^j - \sum_{k \in \mathcal{N}} m_k (A^{0i} \hat{a}^{ik})^\sim (Q_1^k + Q_{IV}^k + Q_V^k) \tag{84}$$

The above equation is a simplified version of equation 11.1.5-25b in [19]. This simplification was possible because here the main reference frame is an inertial reference frame and therefore not subject to any acceleration.

In equation 84, the tilde operator, \sim , has the following meaning. If u is a 3×1 column matrix, such that $u = [u_1 \ u_2 \ u_3]^T$, then \tilde{u} or $(u)^\sim$ is defined to be the 3×3 skew-symmetric matrix shown below:

$$\tilde{u} = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \tag{2.2.1-10} \tag{85}$$

Also in equation 84, it can be shown (using the definitions in [19]) that the terms Q_I^i and Q_V^i are zero for this problem. The reasoning behind $Q_I^i = 0$ is that, for each bar, the vectors linking its COM to the attachment points of the joints incident on that bar are constant when resolved in the corresponding body-fixed frame. In addition, the reason why $Q_V^i = 0$ for the double pendulum problem is that there is no translational motion across either joint.

The remaining Q terms on the RHS of equation 84 are Q_{II}^i , Q_{III}^i and Q_{IV}^i . The first of these, Q_{II}^i , is defined by the following equation:

$$Q_{II}^i = A^{0i} \tilde{\omega}^i I^i \omega^i \tag{86}$$

where ω^i is the absolute angular velocity of body i resolved in the frame fixed in that body. In this case:

$$\omega^1 = [0 \quad 0 \quad \dot{\theta}_1]^T \quad \text{and} \quad \omega^2 = [0 \quad 0 \quad \dot{\theta}_1 + \dot{\theta}_2]^T \tag{87}$$

Equation 86 is a simplification of equations 11.1.5-24b and 11.1.5-22b in [19]. The justification for this simplification is that in the double pendulum problem the inertia matrices, I^1 and I^2 , are constant (see equation 43), and the bars are not gyrostats but just rigid bodies. By using equations 43, 85 and 87 in equation 86, it can be shown that Q_{II}^i is zero for this problem.

The second of the remaining Q terms on the RHS of equation 84 is Q_{III}^i . This term is defined as follow:

$$Q_{III}^i = \sum_{a \in \mathcal{A}} T^{ai} A^{0f(a)} \tilde{\Omega}^a \omega^{f(a)} \tag{11.1.5-24c} \tag{88}$$

where, as stated previously, Ω^a contains the components of the relative angular velocity across joint a with respect to the frame fixed in the body from which that joint emanates. In this case:

$$\Omega^{a1} = [0 \quad 0 \quad \dot{\theta}_1]^T \quad \text{and} \quad \Omega^{a2} = [0 \quad 0 \quad \dot{\theta}_2]^T \tag{89}$$

If equation 88 is expanded for the double pendulum problem it can be seen that:

$$Q_{III}^i = T^{ai} A^{00} \tilde{\Omega}^{a1} \omega^0 + T^{a2i} A^{01} \tilde{\Omega}^{a2} \omega^1 \tag{90}$$

In the above equation, the first term on the RHS is zero because the absolute angular velocity of the inertial reference frame, ω^0 , is zero. Furthermore, by using equations 85, 87 and 89 it can be shown that the second term on the RHS of the above equation is also zero. Therefore, the term Q_{III}^i is zero for the present problem.

By combining all the information from the above analysis it is evident that Q_{IV}^i is the only Q term left on the RHS of equation 84, and so that equation become:

$$Q_I''^i = - \sum_{k \in \mathcal{N}} m_k (A^{0i} \hat{a}^{ik})^\sim Q_{IV}^k \tag{91}$$

which, when expanded, yields:

$$Q_I''^1 = -m_1 (A^{01} \hat{a}^{11})^\sim Q_{IV}^1 - m_2 (A^{01} \hat{a}^{12})^\sim Q_{IV}^2 \tag{92}$$

and

$$Q_I''^2 = -m_1 (A^{02} \hat{a}^{21})^\sim Q_{IV}^1 - m_2 (A^{02} \hat{a}^{22})^\sim Q_{IV}^2 \tag{93}$$

Therefore, in order to calculate $Q_I''^1$ and $Q_I''^2$, it is necessary to calculate Q_{IV}^1 and Q_{IV}^2 . The definition of Q_{IV}^i is given by the following equation:

$$Q_{IV}^i = \sum_{j \in \mathcal{O}} A^{0j} (\omega^{jT} \omega^j E - \omega^j \omega^{jT}) \hat{a}^{ji} \tag{11.1.5-24d} \tag{94}$$

and this means that, here:

$$Q_{IV}^1 = A^{01} (\omega^{1T} \omega^1 E - \omega^1 \omega^{1T}) \hat{a}^{11} + A^{02} (\omega^{2T} \omega^2 E - \omega^2 \omega^{2T}) \hat{a}^{21} \tag{95}$$

and

$$Q_{IV}^2 = A^{01}(\omega^{1T} \omega^1 E - \omega^1 \omega^{1T}) \hat{a}^{12} + A^{02}(\omega^{2T} \omega^2 E - \omega^2 \omega^{2T}) \hat{a}^{22} \tag{96}$$

By substituting equations 41, 52–55 and 87 into equations 95 and 96 and performing some simple algebra it can be shown that, for the present problem:

$$Q_{IV}^1 = \frac{l_1}{2} \dot{\theta}_1^2 \begin{bmatrix} -\sin \theta_1 \\ \cos \theta_1 \\ 0 \end{bmatrix} \tag{97}$$

and

$$Q_{IV}^2 = l_1 \dot{\theta}_1^2 \begin{bmatrix} -\sin \theta_1 \\ \cos \theta_1 \\ 0 \end{bmatrix} + \frac{l_1}{2} (\dot{\theta}_1 + \dot{\theta}_2)^2 \begin{bmatrix} -\sin(\theta_1 + \theta_2) \\ \cos(\theta_1 + \theta_2) \\ 0 \end{bmatrix} \tag{98}$$

Finally, $Q_1'^1$ and $Q_1'^2$ can be obtained for the double pendulum problem by using equations 64–67, 85, 97 and 98 in equations 92 and 93. After some more algebra this gives:

$$Q_1'^1 = -\frac{m_2 l_1 l_2}{2} (\dot{\theta}_1 + \dot{\theta}_2)^2 \begin{bmatrix} 0 \\ 0 \\ \sin \theta_2 \end{bmatrix} \tag{99}$$

and

$$Q_1'^2 = \frac{m_2 l_1 l_2}{2} \dot{\theta}_1^2 \begin{bmatrix} 0 \\ 0 \\ \sin \theta_2 \end{bmatrix} \tag{100}$$

Calculating the $Q_i'^1$ and $Q_i'^2$ terms

At this point it is appropriate to consider the remaining two of the four terms in equations 82 and 83 which are used to calculate χ^{a_1} and χ^{a_2} , namely the terms $Q_1'^1$ and $Q_1'^2$. These terms are defined by the following equation:

$$Q_i'^i = -A^{0i} L^i + \sum_{j \in \mathcal{N}} (A^{0i} \hat{a}^{ij})^{\sim} A^{0j} F^j \tag{11.1.5-26b} \tag{101}$$

where L^i and F^i are the externally applied torque and force on body i , respectively, and these are resolved in the frame fixed in body i . As a result, $A^{0i} L^i$ and $A^{0i} F^i$ are these quantities resolved in the inertial reference frame (frame 0). In this case there are no externally applied torques on either body (i.e. $L^i = 0$ for $i = 1$ and 2) but:

$$A^{0i} F^i = \begin{bmatrix} 0 \\ -m_i g \\ 0 \end{bmatrix} \tag{102}$$

Therefore, in this case:

$$Q_1^{a1} = (A^{01}\hat{a}^{11})^\sim A^{01}F^1 + (A^{01}\hat{a}^{12})^\sim A^{02}F^2 = \left(\frac{m_1}{2} + m_2\right)gl_1 \begin{bmatrix} 0 \\ 0 \\ \sin\theta_1 \end{bmatrix} \quad (103)$$

and

$$Q_1^{a2} = (A^{02}\hat{a}^{21})^\sim A^{01}F^1 + (A^{02}\hat{a}^{22})^\sim A^{02}F^2 = \frac{1}{2}m_2gl_2 \begin{bmatrix} 0 \\ 0 \\ \sin(\theta_1 + \theta_2) \end{bmatrix} \quad (104)$$

Equations 103 and 104 were obtained using equations 64–67.

Final construction of χ_1^a and χ_2^a

It is now possible to calculate χ^{a1} and χ^{a2} . Substituting equations 99, 100, 103 and 104 into equations 82 and 83 gives:

$$\begin{aligned} \chi^{a1} = & \frac{m_2l_1l_2}{2}(\dot{\theta}_1 + \dot{\theta}_2)^2 \sin\theta_2 - \left(\frac{m_1}{2} + m_2\right)gl_1 \sin\theta_1 \\ & - \frac{m_2l_1l_2}{2}\dot{\theta}_1^2 \sin\theta_2 - \frac{m_2gl_2}{2}\sin(\theta_1 + \theta_2) \end{aligned} \quad (105)$$

and

$$\chi^{a2} = -\frac{m_2l_1l_2}{2}\dot{\theta}_1^2 \sin\theta_2 - \frac{m_2gl_2}{2}\sin(\theta_1 + \theta_2) \quad (106)$$

The λ^a term

The last term in equation 28 is λ^a and it is necessary to calculate this term for $a = a_1$ and $a = a_2$. From equation 31 states that:

$$\lambda^a = \phi^{aT} \Lambda^a$$

where, as stated previously, Λ^a is the 6×1 matrix which contains the two 3×1 submatrices L^a and F^a (see equation 22). The quantities L^a and F^a are the components of the interaction torque and force, respectively, across joint a with respect to the frame fixed in the body from which that joint emanates.

Since both joints in the double pendulum are revolute joints, a very similar argument to that used in the example above (for free mode vectors) can easily be applied to show that:

$$\lambda^{a1} = \lambda^{a2} = 0 \quad (107)$$

The construction of the equations of motion

Finally, the equations of motion for the double pendulum can be formally constructed. Setting $a = a_1$ in equation 28 and using equations 35, 72, 73, 105 and 107, the a_1 equation of motion is obtained:

$$\begin{aligned}
& \left(\left(\frac{m_1}{3} + m_2 \right) l_1^2 + m_2 l_1 l_2 \cos \theta_2 + \frac{m_2 l_2^2}{3} \right) \ddot{\theta}_1 + \left(\frac{m_2 l_1 l_2}{2} \cos \theta_2 + \frac{m_2 l_2^2}{3} \right) \ddot{\theta}_2 \\
& = \frac{m_2 l_1 l_2}{2} (\dot{\theta}_1 + \dot{\theta}_2)^2 \sin \theta_2 - \left(\frac{m_1}{2} + m_2 \right) g l_1 \sin \theta_1 \\
& \quad - \frac{m_2 l_1 l_2}{2} \dot{\theta}_1^2 \sin \theta_2 - \frac{m_2 g l_2}{2} \sin(\theta_1 + \theta_2) \tag{108}
\end{aligned}$$

Setting $a = a_2$ in equation 28 and using equations 35, 74, 75, 106 and 107, the a_2 equation of motion is stated as follows:

$$\left(\frac{m_2 l_1 l_2}{2} \cos \theta_2 + \frac{m_2 l_2^2}{3} \right) \ddot{\theta}_1 + \frac{m_2 l_2^2}{3} \ddot{\theta}_2 = - \frac{m_2 l_1 l_2}{2} \dot{\theta}_1^2 \sin \theta_2 - \frac{m_2 g l_2}{2} \sin(\theta_1 + \theta_2) \tag{109}$$

Applications beyond the double pendulum problem

The Roberson and Schwertassek method can be applied to much more complicated problems than just the double pendulum problem. It can be applied to any multi-body system made up of rigid and gyrostatic bodies and for which the joint free mode vectors can be derived. The free mode vectors can be easily obtained for the vast majority mechanical joints. For example, free mode vectors can be readily calculated for prismatic (i.e. linear), universal and spherical joints. Furthermore, the Roberson and Schwertassek method can also be applied to multibody systems containing non-holonomic joints, such as rolling/joints.

If the Roberson and Schwertassek method is applied to an open-loop multibody system, then, as described earlier, the state-space equations of motion will be automatically derived. If, on the other hand, the Roberson and Schwertassek method is applied to a closed-loop multibody system, then the resultant equations of motion will be in differential algebraic form. It was explained above that it is impossible to obtain the state-space equations of motion for a general three-dimensional closed-loop multibody system. Fortunately, for a closed-loop system, the Roberson and Schwertassek method will yield a relatively *small* set of differential algebraic equations of motion, which, consequently, are ‘nearly’ in state-space form.

Summary and conclusions

At the beginning of this paper a brief review of multibody methods was presented, together with an explanation of the motivation for the current work. A multibody method is an algorithm which automatically derives the equations of motion for a system of rigid or flexible bodies connected by joints. In most cases the multibody method then goes on to solve these equations of motion for the response of the system.

In this study, the multibody method developed by Roberson and Schwertassek [19] was chosen for investigation. This is a classical Eulerian multibody code and the book which describes it [19] is very highly regarded by multibody experts. Roberson and Schwertassek’s method can automatically obtain the equations of

motion for systems of rigid and gyrostatic bodies. The algorithm uses relative rather than absolute coordinates and it can also handle non-holonomic constraints.

The present paper has given a description of how the Roberson and Schwertassek method uses graph theory in order to describe the interconnectedness of the bodies. It went on to explain that the Roberson and Schwertassek algorithm reformulates the unreduced dynamic equations of motion in terms of relative (or joint) coordinates. Then, in the following section, it described how, for systems with a tree structure, the algorithm reduces these equations to state–space form by using the free mode vectors associated with the joints. Finally, the Roberson and Schwertassek method was applied to the double pendulum problem in order to obtain its equations of motion in state–space form.

The main goal of all the above is to allow the reader rapidly to acquire a relatively deep understanding of the Roberson and Schwertassek method, which would not be as easy by reading their long book [19]. To this end, this paper has presented a concise but detailed description of the method, followed by a *fully* worked didactic example of the application of the multibody code to a practical problem. Such worked examples are rare in the literature and indeed there are no worked examples in Roberson and Schwertassek's study [19].

This paper represents part of a larger research project, the aim of which is to develop a *fully symbolic analytical* multibody code. To the authors' knowledge, most, if not all, multibody codes which go on to solve the resultant equations of motion do so numerically. Consequently, in this project, the authors have implemented an open-loop formulation of Roberson and Schwertassek's method in the Mathematica software program. As stated earlier, the *symbolic* equations of motion which result from the above code will be in state–space form and therefore approximate analytical solutions to these equations can be sought by applying the *multiple scales* method. This method is a *perturbation* method and it has also been encoded in Mathematica by Khanin *et al.* [20–22]. In summary, the authors believe that this represents the first fully symbolic analytical multibody formulation, and they present the methodology in this form in the hope that others will be motivated to assimilate and apply it.

Acknowledgements

The authors would like to thank Mr A. A. B. MacLean of W. S. Atkins Ltd (Aberdeen) for his support during this project. The authors would also like to acknowledge the EPSRC grants, GR/N32280/01 and GR/N32334, which have enabled this research to go ahead.

References

- [1] E. J. Kreuzer and W. O. Schiehlen, 'Computerized generation of symbolic equations of motion for spacecraft', *Journal of Guidance, Control, and Dynamics*, **8**(2) (1985), 284–287.
- [2] W. O. Schiehlen, 'Computational dynamics of multibody systems', in *Proceedings of ASME International Computers in Engineering Conference and Exposition* (Boston, MA, 1990), pp. 519–525.

- [3] G. C. Andrews, M. J. Richard and R. J. Anderson, 'A general vector-network formulation for dynamic systems with kinematic constraints', *Mechanism and Machine Theory*, **23**(3) (1988), 243–256.
- [4] M. J. Richard, R. J. Anderson and G. C. Andrews, 'The vector-network method for the modeling of mechanical systems', *Mathematics and Computers in Simulation*, **31**(6) (1990), 565–581.
- [5] J. Wittenburg, *Dynamics of Systems of Rigid Bodies* (B. G. Teubner, Stuttgart, 1977).
- [6] J. Wittenburg, 'Graph-theoretical methods in multibody dynamics', *Contemporary Mathematics*, **97** (1989), 459–468.
- [7] N. Orlandea, M. A. Chace and D. A. Calahan, 'A sparsity-oriented approach to the dynamic analysis and design of mechanical systems – parts 1 and 2', *ASME Journal of Engineering for Industry*, **99** (1977), 773–784.
- [8] M. A. Chace, 'Methods and experience in computer aided design of large displacement mechanical systems', in E. J. Haug (ed.), *Computer Aided Analysis and Optimization of Mechanical System Dynamics* (Springer, Berlin, 1984), p. 233.
- [9] R. A. Wehage and E. J. Haug, 'Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems', *Journal of Mechanical Design*, **104**(1) (1982), 247–255.
- [10] P. N. Sheth and J. J. Uicker, 'IMP (Integrated Mechanisms Program), a computer-aided design analysis system for mechanisms and linkage', *ASME Journal of Engineering for Industry*, **94** (1972), 454–464.
- [11] A. Avello, J. G. de Jalón and E. Bayo, 'Dynamics of flexible multibody systems using Cartesian coordinates and large displacement theory', *International Journal for Numerical Methods in Engineering*, **32**(8) (1991), 1543–1563.
- [12] A. A. Shabana, *Dynamics of Multibody Systems*, 2nd edn (Cambridge University Press, Cambridge, 1998).
- [13] J. J. McPhee, 'On the use of linear graph theory in multibody system dynamics', *Nonlinear Dynamics*, **9** (1996), 73–90.
- [14] N. Christofides, *Graph Theory: An Algorithmic Approach* (Academic Press, New York, 1975).
- [15] P. Shi and J. McPhee, 'Dynamics of flexible multibody systems using virtual work and linear graph theory', *Multibody System Dynamics*, **4**(4) (2000), 355–381.
- [16] P. Fisette, D. A. Johnson and J. C. Samin, 'A fully symbolic generation of the equations of motion of multibody systems containing flexible beams', *Computer Methods in Applied Mechanics and Engineering*, **142** (1997), 123–152.
- [17] F. Melzer, 'Symbolic computations in flexible multibody systems', *Nonlinear Dynamics*, **9** (1996), 147–163.
- [18] D. I. M. Forehand, R. Khanin and M. P. Cartmell, 'A Lagrangian multibody code for deriving the symbolic state–space equations of motion for open-loop systems containing flexible beams', *Mathematics and Computers in Simulation* (in press).
- [19] R. E. Roberson and R. Schwertassek, *Dynamics of Multibody Systems* (Springer-Verlag, Berlin, 1988).
- [20] R. Khanin and M. P. Cartmell, 'Applying the method of multiple-scales', *Mathematica in Education and Research*, **8**(2) (1999), 19–26.
- [21] R. Khanin, M. P. Cartmell and A. Gilbert, 'A computerised implementation of the multiple scales perturbation method using Mathematica', *Computers and Structures*, **76**(5) (2000), 565–575.
- [22] R. Khanin and M. P. Cartmell, 'Parallelisation of perturbation analysis: application to large-scale engineering problems', *Journal of Symbolic Computation*, **31** (2001), 461–473.