A Review of Multibody Dynamics Formulations

D. de Falco [1] D. Guida [2] E. Pennestrì[1]

[1]Università di Napoli - Italy
[2]Università di Salerno - Italy
[3]Università di Roma Tor Vergata - Italy
Dipartimento di Ingegneria Meccanica

First AAM 2008, New Orleans 2008

Presentation outline

- Review of *classic* multibody dynamics formulations.
- Compare the results of numerical tests on the reviewed and two new dynamic formulations.

Presentation outline

- Review of *classic* multibody dynamics formulations.
- Compare the results of numerical tests on the reviewed and two new dynamic formulations.

The set of coordinates in multibody dynamics

1. Minimal set of coordinates (ODE)
2. Redundant set of coordinates (DAE)

The set of coordinates in multibody dynamics

1. Minimal set of coordinates (ODE)
2. Redundant set of coordinates (DAE)

The multibody equations of dynamics

Differential-Algebraic Equations (DAE) system (differential index 3)

$$[M]\{\ddot{q}\} + \left[\Psi_q^T\right]\{\lambda\} = \{F_e\} \tag{1a}$$

$$\{\Psi\} = 0 \tag{1b}$$

The multibody equations of dynamics

Differential-Algebraic Equations (DAE) system (differential index 1)

$$[M]\{\ddot{q}\} + \left[\Psi_q^T\right]\{\lambda\} = \{F_e\} \tag{2a}$$

$$[\Psi_q]\{\ddot{q}\} = \{\gamma\} \tag{2b}$$

Common problems

1. Positions and velocity constraints are not necessarily satisfied by the computed numerical solution (solution drift).
2. The constraints equations may not be independent. (*i.e.* Jacobian without a full rank).
3. The mass matrix could be singular.

Common problems

1. Positions and velocity constraints are not necessarily satisfied by the computed numerical solution (solution drift).
2. The constraints equations may not be independent. (*i.e.* Jacobian without a full rank).
3. The mass matrix could be singular.

Common problems

1. Positions and velocity constraints are not necessarily satisfied by the computed numerical solution (solution drift).
2. The constraints equations may not be independent. (*i.e.* Jacobian without a full rank).
3. The mass matrix could be singular.

Numerical integration procedures of multibody dynamics DAE systems

1. Coordinate partitioning method.
2. Maggi like methods. (Orthogonalization of constraints).
3. Udwadia-Kalaba dynamic formulation based on the Gauss' Principle of Least Action .
4. Dynamic formulations deduced from the least squares criterion.

Numerical integration procedures of multibody dynamics DAE systems

1. Coordinate partitioning method.
2. Maggi like methods. (Orthogonalization of constraints).
3. Udwadia-Kalaba dynamic formulation based on the Gauss' Principle of Least Action .
4. Dynamic formulations deduced from the least squares criterion.

Numerical integration procedures of multibody dynamics DAE systems

1. Coordinate partitioning method.
2. Maggi like methods. (Orthogonalization of constraints).
3. Udwadia-Kalaba dynamic formulation based on the Gauss' Principle of Least Action .
4. Dynamic formulations deduced from the least squares criterion.

Numerical integration procedures of multibody dynamics DAE systems

1. Coordinate partitioning method.
2. Maggi like methods. (Orthogonalization of constraints).
3. Udwadia-Kalaba dynamic formulation based on the Gauss' Principle of Least Action .
4. Dynamic formulations deduced from the least squares criterion.

The coordinate partitioning method

The set of coordinates is partitioned as follows:

$$\{q\}_{n\times 1} = \begin{Bmatrix} v \\ u \end{Bmatrix} \tag{3}$$

where

- $\{u\}_{(n-F)\times 1}$=**dependent** coordinates
- $\{v\}_{F\times 1}$=**independent** coordinates

The coordinate partitioning method

Equation of dynamics in partitioned form (Haug, Wehage, 1982)

$$[M^{vv}]\{\ddot{v}\} + [M^{vu}]\{\ddot{u}\} + [\Psi_v]^T\{\lambda\} = \{Q^v\} \quad (4)$$

$$[M^{uv}]\{\ddot{v}\} + [M^{uu}]\{\ddot{u}\} + [\Psi_u]^T\{\lambda\} = \{Q^u\} \quad (5)$$

$$[\Psi_u]_{m\times(n-F)}\{\ddot{u}\} + [\Psi_v]_{m\times F}\{\ddot{v}\} = \{\gamma\} \quad (6)$$

The coordinate partitioning method

Eliminating λ and $\ddot{u}$ one obtains

$$\left[\hat{M}\right]_{F\times F}\{\ddot{v}\}_{F\times 1}=\left\{\hat{Q}\right\}_{F\times 1} \tag{7}$$

where

$$\begin{aligned}\left[\hat{M}\right]&=[M^{vv}]-[M^{vu}]\,[\Psi_u]^{-1}\,[\Psi_v]\\ &-[\Psi_v]^T\left([\Psi_u]^{-1}\right)^T\left([M^{uv}]-[M^{uu}]\,[\Psi_u]^{-1}\,[\Psi_v]\right)\end{aligned} \tag{8}$$

and

$$\begin{aligned}\left\{\hat{Q}\right\}&=\{Q^v\}-[M^{vu}]\,[\Psi_u]^{-1}\,\{\gamma\}\\ &-[\Psi_v]^T\left([\Psi_u]^{-1}\right)^T\left(\{Q^u\}-[M^{uu}]\,[\Psi_u]^{-1}\,\{\gamma\}\right)\end{aligned} \tag{9}$$

Maggi like methods - Orthogonalization of constraints

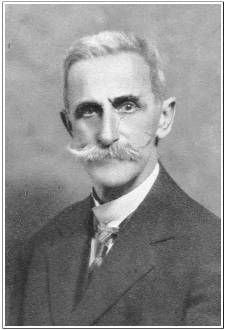


Figure: Gian Antonio Maggi, Milano (1856-1937)

$$[M]\{\ddot{q}\} + [\Psi_q]^T\{\lambda\} = \{Q\} \tag{10}$$

The idea behind the orthogonalization of constraints is the elimination of reaction forces.

For this purpose we need a matrix $[V]$ such that:

$$\{\dot{q}\} = [V]\{\dot{v}\}$$

$$[\Psi_q][V] = [0]$$

$$[M]\{\ddot{q}\} + [\Psi_q]^T\{\lambda\} = \{Q\} \tag{10}$$

The idea behind the orthogonalization of constraints is the elimination of reaction forces.

For this purpose we need a matrix $[V]$ such that:

$$\{\dot{q}\} = [V]\{\dot{v}\}$$

$$[\Psi_q][V] = [0]$$

Since

$$\{\ddot{q}\} = [V]\{\ddot{v}\} + \left[\dot{V}\right]\{\dot{v}\}$$

the equation of dynamics $[M]\{\ddot{q}\} + [\Psi_q]^T\{\lambda\} = \{Q\}$ are transformed into

$$[V]^T[M][V]\{\ddot{v}\} = [V]^T\{Q\} + [V]^T[M]\left[\dot{V}\right]\{\dot{v}\} \qquad (11)$$

How to compute $\left[\dot{V}\right]\{\dot{v}\}$

Transform $[\Psi_q]\{\ddot{q}\} = \{\gamma\}$ into

$$[\Psi_u]\{\ddot{u}\} + [\Psi_v]\{\ddot{v}\} = \{\gamma_u\} \tag{12}$$

$$\{\ddot{q}\} = \left\{ \begin{array}{c} \ddot{u} \\ \ddot{v} \end{array} \right\} = \left[\begin{array}{c} -[\Psi_u]^{-1}[\Psi_v] \\ [I] \end{array} \right]\{\ddot{v}\} + \left\{ \begin{array}{c} [\Psi_u]^{-1}\{\gamma_u\} \\ \{0\} \end{array} \right\} \tag{13}$$

How to compute $\left[\dot{V}\right]\{\dot{v}\}$

Transform $[\Psi_q]\{\ddot{q}\} = \{\gamma\}$ into

$$[\Psi_u]\{\ddot{u}\} + [\Psi_v]\{\ddot{v}\} = \{\gamma_u\} \quad (12)$$

$$\{\ddot{q}\} = \left\{ \begin{array}{c} \ddot{u} \\ \ddot{v} \end{array} \right\} = \left[\begin{array}{c} -[\Psi_u]^{-1}[\Psi_v] \\ [I] \end{array} \right] \{\ddot{v}\} + \left\{ \begin{array}{c} [\Psi_u]^{-1}\{\gamma_u\} \\ \{0\} \end{array} \right\} \quad (13)$$

1. The set $\{v\}$ of independent coordinates and the matrix $[V]$ is not unique.
2. There are different numerical methods for the computation of matrix $[V]$

Numerical methods for computing the matrix $[V]$

1. Zero Eigenvalue method (Walton and Steeves, 1969)
2. QR decomposition (Kim and Vanderploeg, 1986)
3. Singular value decomposition (SVD), (Mani and Haug, 1985), Likins and Singh (1985)
4. The PUTD decomposition (Amirouche et al., 1988)
5. The QTZ decomposition (Vita and Pennestrì, 2004)
6. The Schur decomposition (Pennestrì and Valentini, 2007)

Zero Eigenvalue method - Walton and Steeves (1969), Kamman and Huston (1985)

Let us form the matrix

$$[H] = [\Psi_q]^T [\Psi_q] \tag{14}$$

Since $[H]_{n \times n}$ is symmetric, all the eigenvalues are positive and there exists the similarity transform

$$[T]^T [H] [T] = [\Lambda] \tag{15}$$

where $[T]$ is the orthogonal matrix of eigenvectors and $[\Lambda]$ the diagonal matrix of eigenvalues.

The matrix $[T]$ can be partitioned as follows

$$[T] = \begin{bmatrix} [T_1]_{n\times(n-r)} & [T_2]_{n\times r} \end{bmatrix} \tag{16}$$

where $[T_1]$ is the matrix of the eigenvectors associated to zero eigenvalues.

It can be demonstrated the condition of orthogonality

$$[\Psi_q]\,[T_1] = [0] \tag{17}$$

$$\therefore [V] = [T_1]$$

The matrix $[T]$ can be partitioned as follows

$$[T] = \left[\; [T_1]_{n \times (n-r)} \quad [T_2]_{n \times r} \; \right] \tag{16}$$

where $[T_1]$ is the matrix of the eigenvectors associated to zero eigenvalues.
It can be demonstrated the condition of orthogonality

$$[\Psi_q]\,[T_1] = [0] \tag{17}$$

$$\therefore [V] = [T_1]$$

QR decomposition - Kim and Vanderploeg (1986)

$$[\Psi_q]^T = \begin{bmatrix} [Q_1]_{n\times(n-F)} & [Q_2]_{n\times F} \end{bmatrix} \begin{bmatrix} [R_1]_{r\times m} \\ [0]_{F\times m} \end{bmatrix} = [Q_1]\,[R_1] \tag{18}$$

where $[Q_1]$ and $[Q_2]$ simultaneously satisfy the conditions

$$[Q_2]^T\,[Q_1] = [0] \tag{19}$$

$$[Q_2]^T\,[Q_2] = [I] \tag{20}$$

QR decomposition (cont.)

$$[Q_2]^T [\Psi_q]^T = [Q_2]^T [Q_1] [R_1] = [0] \tag{21}$$

$$\therefore [V] = [Q_2]$$

SVD method - Singh and Likins (1985), Mani and Haug (1985)

$$[\Psi_q]^T = [W][\Lambda][U] \quad (22)$$

$$\left[\begin{array}{cc} [W_d]_{n\times r} & [W_i]_{n\times(n-r)} \end{array}\right] \left[\begin{array}{c} [\Lambda_1]_{r\times r} \\ [0]_{(n-r)\times r} \end{array}\right] [U]^T_{r\times r}$$

$$= [W_d][\Lambda_1][U]^T \quad (23)$$

where $[W_i]^T [W_d] = [0]$

SVD method (cont.)

$$[W_i]^T [\Psi_q]^T = [W_i]^T [W_d] [\Lambda_1] [U]^T = [0] \tag{24}$$

$$\therefore [V] = [W_i]$$

PUTD method - Amirouche *et al.* (1988)

The method's first step is to obtain the Householder transform matrix $[H]$ of $[\Psi_q]^T$.
Then, through the Gram-Schmidt process, it is identified a matrix $[D_2]$ such that

$$[D_2]_{(n-r)\times n}\,[H]^T_{n\times n}\,[\Psi_q]^T = [0] \qquad (25)$$

$$\therefore [V] = [H]\,[D_2]^T$$

The matrix $[D_2]$ has the following block structure

$$[D_2] = \left[\begin{array}{cc} [0]_{(n-r)\times r} & [I]_{(n-r)\times(n-r)} \end{array} \right] \tag{26}$$

One can verify that $[D_2]\,[H]^T = [Q_2]^T$.

Schur decomposition - (Pennestrì and Valentini, 2007)

The Schur decomposition of the matrix $[H]$ has the form

$$[K]^T [H] [K] = [U] \tag{27}$$

where $[K]$ is a unitary matrix and $[U]$ an upper triangular matrix. The eigenvalues of $[H]$ are on the diagonal of $[U]$.
The matrix $[K]$ can be partitioned as follows

$$[K] = \left[\begin{array}{cc} [K_1]_{n\times(n-r)} & [K_2]_{n\times r} \end{array} \right] \tag{28}$$

where $[K_1]$ are the generalized Schur vectors associated to zero eigenvalues.

Schur decomposition - (Pennestrì and Valentini, 2007)

The Schur decomposition of the matrix $[H]$ has the form

$$[K]^T [H] [K] = [U] \tag{27}$$

where $[K]$ is a unitary matrix and $[U]$ an upper triangular matrix.
The eigenvalues of $[H]$ are on the diagonal of $[U]$.
The matrix $[K]$ can be partitioned as follows

$$[K] = \left[\begin{array}{cc} [K_1]_{n\times(n-r)} & [K_2]_{n\times r} \end{array} \right] \tag{28}$$

where $[K_1]$ are the generalized Schur vectors associated to zero eigenvalues.

Schur decomposition (cont.)

It can be easily verified the condition of orthogonality

$$[\Psi_q]\,[K_1] = [0] \tag{29}$$

$$\therefore\ [V] = [K_1]$$

Udwadia-Kalaba fundamental equation (1996)

From Gauss' Least Action Principle one obtains

$$\{\ddot{q}\} = \{\ddot{q}_f\} + [M]^{-\frac{1}{2}} [D]^{+} (\{\gamma\} - [\Psi_q] \{\ddot{q}_f\}) \ . \qquad (30)$$

where

$$\{\ddot{q}_f\} = [M]^{-1} \{F\} \ , \qquad (31)$$

and

$$[D] = [\Psi_q] [M]^{-\frac{1}{2}} \ , \qquad (32)$$

with

$$[M]^{-1} = [M]^{-\frac{1}{2}} [M]^{-\frac{1}{2}} \ . \qquad (33)$$

Udwadia-Phohomsiri equation (2006)

$$\{\ddot{q}\} = \left[\,\overline{M}\,\right]^{+} \left\{ \begin{array}{c} F \\ \gamma \end{array} \right\} \tag{34}$$

where

$$\left[\,\overline{M}\,\right] = \left[\begin{array}{c} \left(I - \Psi_q^{+}\Psi_q\right) M \\ \Psi_q \end{array} \right] \tag{35}$$

Note: It can be used also when the mass matrix is singular.

$$\begin{bmatrix} M & \Psi_q^T \\ \Psi_q & 0 \end{bmatrix} \begin{Bmatrix} \ddot{q} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} Q \\ \gamma \end{Bmatrix} \tag{36}$$

de Falco, Pennestrì, Vita - Formulation 1

The pseudoinverse of the matrix of coefficients in the equation of dynamics can be expressed in the form

$$\begin{bmatrix} M & \Psi_q^T \\ \Psi_q & 0 \end{bmatrix}^+ = \begin{bmatrix} M^{-1} - M^{-1}\Psi_q^T Q^+ \Psi_q M^{-1} & M^{-1}\Psi_q^T Q^+ \\ Q^+ \Psi_q M^{-1} & -Q^+ \end{bmatrix} . \tag{37}$$

where

$$[Q] = [\Psi_q] [M]^{-1} [\Psi_q]^T . \tag{38}$$

de Falco, Pennestrì, Vita - Formulation 1

$$\{\ddot{q}\} = ([I] - [H][\Psi_q])[M]^{-1}\{F\} + [H]\{\gamma\} \tag{39}$$

where

$$[H] = [M]^{-1}[\Psi_q]^T [Q]^+ \ , \tag{40}$$

$$[Q] = [\Psi_q][M]^{-1}[\Psi_q]^T \ . \tag{41}$$

$$\begin{bmatrix} M & \Psi_q^T \\ \Psi_q & 0 \end{bmatrix}^+ = \begin{bmatrix} 0 & \Psi_q^+ \\ \left([\Psi_q]^+\right)^T & -\left([\Psi_q]^+\right)^T [R] \end{bmatrix} + \begin{bmatrix} I \\ -R^T \end{bmatrix} [Q] \begin{bmatrix} I & -R \end{bmatrix} .$$

where

$$[E] = [I] - [\Psi_q]^+ [\Psi_q] \ , \tag{42}$$

$$[Q] = ([E]\,[M]\,[E])^+ \ , \tag{43}$$

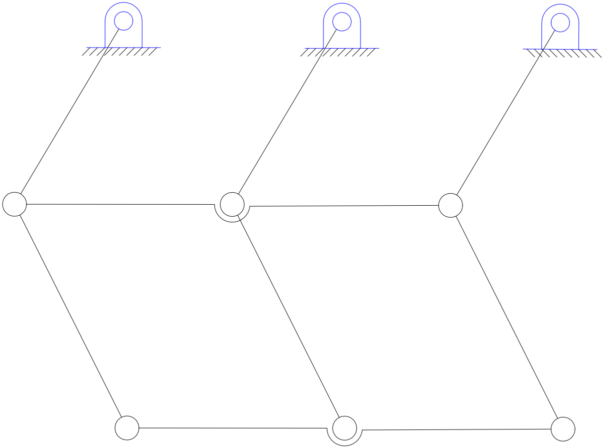
$$[R] = [M]\,[\Psi_q]^+ \ , \tag{44}$$

de Falco, Pennestrì, Vita - Formulation 2

$$\{\ddot{q}\} = [Q]\{F\} + \left([\Psi_q]^+ - [Q][M][\Psi_q]^+\right)\{\gamma\} \ . \tag{45}$$

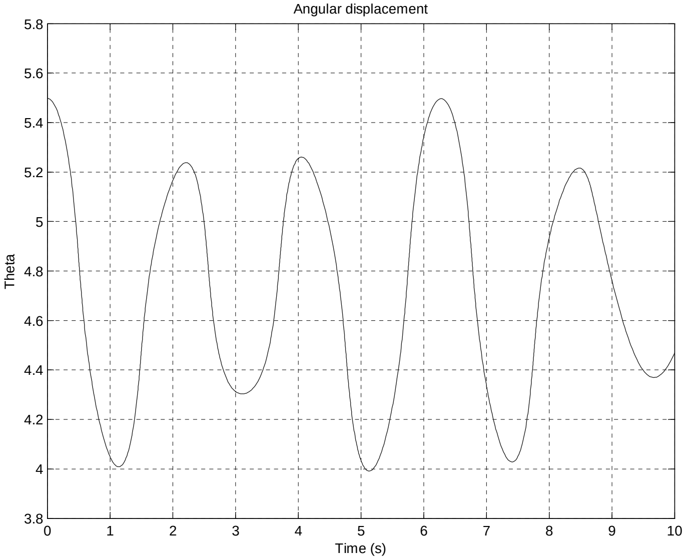
Note: It can be used also when the mass matrix is singular.

The mechanical system analyzed



Problem dimension

- Generalized coordinates - $\{q\}$: 24
- Mass matrix - $[M]$: 24 $\times$ 24
- Jacobian matrix - $[\Psi_q]$: 24 $\times$ 24 (rank: 22)
- Force vector $\{F\}$: 24


Angular displacement
Theta
5.8
5.6
5.4
5.2
5
4.8
4.6
4.4
4.2
4
3.8
0
1
2
3
4
5
6
7
8
9
10
Time (s)

Programming info

- All computations using standard MATLAB libraries
- In the subroutines only array operations have been used.
- Numerical integration by means of `ODE45` with `RelTol`=10^{-6}, `AbsTol`=10^{-6}
- Rank and Gaussian elimination using 10^{-6} as a treshold value for zero.

Numerical results

Table: Comparison of computational efficiency.

Method	**CPU Time** (s)	Function calls	ODE Solver (s)
Zero Eigenvalue	66.2	1840	0.38
Coordinate partitioning	128	1846	0.47
QR	66.1	1840	0.38
SVD	Fails		
PUTD	66.5	1840	0.38
Schur	66.7	1828	0.38
Udwadia-Kalaba	2.65	1954	0.36
Udwadia-Phomsiri	3.88	1828	0.38
de Falco *et al.* (1)	2.61	1828	0.36
de Falco *et al.* (2)	3.60	1828	0.43

Conclusions

1. A review and comparison of different multibody dynamics formulations has been presented.
2. The numerical accuracy seems not influenced by the method used for the orthogonalization. However, this may not be necessarily true for large scale problems.
3. Rank computation problems.
4. Iterative algorithms for the algebra decompositions should be used.

Conclusions

1. A review and comparison of different multibody dynamics formulations has been presented.
2. The numerical accuracy seems not influenced by the method used for the orthogonalization. However, this may not be necessarily true for large scale problems.
3. Rank computation problems.
4. Iterative algorithms for the algebra decompositions should be used.

Conclusions

1. A review and comparison of different multibody dynamics formulations has been presented.
2. The numerical accuracy seems not influenced by the method used for the orthogonalization. However, this may not be necessarily true for large scale problems.
3. Rank computation problems.
4. Iterative algorithms for the algebra decompositions should be used.

Conclusions

1. A review and comparison of different multibody dynamics formulations has been presented.
2. The numerical accuracy seems not influenced by the method used for the orthogonalization. However, this may not be necessarily true for large scale problems.
3. Rank computation problems.
4. Iterative algorithms for the algebra decompositions should be used.

MATLAB CODE

```
function[ddq]=zeroeigenvalueform(M,Psiq,gamma,f)
% It is assumed that the dynamics equations are in the followi
%      _              _
%     |  M   Psiq^T   | |  ddq    |      |   f    |
%     |               | |         | =
%     |  Psiq   0     | | lambda |      | gamma |
[m,n]=size(Psiq);
[BBg,ipoint1]=rref([Psiq gamma],1.e-3);
r=length(ipoint1);
Psiqm=BBg(1:r,1:n);
[T,S] = eig(Psiqm'*Psiqm);
V=T(1:n,1:(n-r));
BB=V'; gammau=BBg(1:n,n+1);
Sgamma=([Psiqm;BB])\gammau;
Mh=V'*M*V; Fh=V'*f-V'*M*Sgamma;
ddv=Mh\Fh; ddq=V*ddv+Sgamma;
end end of function zeroeigenvalueform
```

```
function[ddq]=svdform(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%           by means of svd formulation (Mani-Haug 1986).
% It is assumed that the dynamics equations are in the followi
%
%      _               _
%     |  M    Psiq^T   | |  ddq    |       |    f    |
%     |                | |         | =
%     |  Psiq   0      | | lambda  |       | gamma  |
[m,n]=size(Psiq);
[BBg,ipoint1]=rref([Psiq gamma],1.e-3);
r=length(ipoint1);
[Uh,S1,Vh] = svd(Psiq');
V=Uh(1:n,r+1:(n));
% Compute [S]*{gamma}
BB=V'; [mbb,nbb]=size(BB); Psiqm=BBg(1:r,1:n);
gammau=BBg(1:n,n+1);
Sgamma=([Psiqm;BB])\gammau;
Mh=V'*M*V; Fh=V'*f-V'*M*Sgamma;
ddv=Mh\Fh; ddq=V*ddv+Sgamma;
end %end of function svdform
```

```
function[ddq]=qrform(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of qr formulation (Kim-Vanderploeg 1986).
% It is assumed that the dynamics equations are in the followi
%       _              _
%      |  M   Psiq^T   | |  ddq    |       |   f    |
%      |               | |         | =
%      |  Psiq   0     | | lambda  |       | gamma  |
[m,n]=size(Psiq);
[BBg,ipoint1]=rref([Psiq gamma],1.e-3);
r=length(ipoint1);
Psiqm=BBg(1:r,1:n);
[QQ,R]=qr(Psiqm');
V=QQ(1:n,r+1:n);
BB=V'; gammau=BBg(1:n,n+1);
Sgamma=([Psiqm;BB])\gammau;
Mh=V'*M*V;
Fh=V'*f-V'*M*Sgamma; ddv=Mh\Fh;
ddq=V*ddv+Sgamma;
end  %end of function qrform
```

```
function[ddq]=putdform(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of putd formulation
% It is assume that the dynamics equations are in the followin
%       _               _
%      |  M   Psiq^T   | |  ddq    |      |   f    |
%      |               | |         | =
%      |  Psiq   0     | | lambda  |      | gamma  |
[m,n]=size(Psiq);
[BBg,ipoint1]=rref([Psiq gamma],1.e-3);
r=length(ipoint1);
Psiqm=BBg(1:r,1:n);
[H,R]=qr(Psiqm');
D22=eye(n-r,n-r);D21=zeros(n-r,r);
D2T=[D21 D22]; V=H*D2T';
BB=V'; gammau=BBg(1:n,n+1);
Sgamma=([Psiqm;BB])\gammau;
Mh=V'*M*V; Fh=V'*f-V'*M*Sgamma;
ddv=Mh\Fh; ddq=V*ddv+Sgamma;
end %end of function putdform
```

```
function[ddq]=udwadiakalaba(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of the Udwadia-Kalaba formulation.
% It is assumed that the dynamics equations are in the followi
%      _            _
%     |  M   Psiq^T  | |  ddq   |       |   f    |
%     |              | |        | =
%     |  Psiq   0    | | lambda |       | gamma  |
%      -            -
v=diag(M);
v=v.^-1;
v1=v.^1/2;
Minv=diag(v);
Minv12=diag(v1);
ddqf=Minv*f;
D=Psiq*Minv12;
Dps=pinv(D,1.e-6);
a1=(gamma-Psiq*ddqf);
ddq=ddqf+Minv12*Dps*a1;
end % end of function udwadiakalaba
```

```
function[ddq]=schurform(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of qr formulation  (Pennestri'-Valentini,
% It is assume that the dynamics equations are in the followin
%      _             _
%     |  M   Psiq^T  | |  ddq    |      |   f    |
%     |              | |         | =
%     |  Psiq   0    | | lambda  |      | gamma  |
[m,n]=size(Psiq);
[BBg,ipoint1]=rref([Psiq gamma],1.e-3);
r=length(ipoint1);
[US,K] = schur(Psiq'*Psiq);
V=US(1:n,1:(n-r)); BB=V';
[mbb,nbb]=size(BB);
Psiqm=BBg(1:r,1:n);
gammau=BBg(1:n,n+1);
Sgamma=([Psiqm;BB])\gammau;
Mh=V'*M*V;
Fh=V'*f-V'*M*Sgamma; ddv=Mh\Fh; ddq=V*ddv+Sgamma;
end  %end of function schurform
```

```
function[ddq]=udwadia2(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of Udwadia-Phohomsiri.
% It is assume that the dynamics equations are in the followin
%      _             _
%     |  M   Psiq^T  | |  ddq   |       |   f   |
%     |              | |        | =
%     |  Psiq   0    | | lambda |       | gamma |
%      -             -
[m,n]=size(Psiq);
Psiqp=pinv(Psiq,1.e-6);
E=(eye(m)-Psiqp*Psiq)*M;
ddq=pinv([E;Psiq],1.e-6)*[f;gamma];
end % of function udwadia2
```

```
function[ddq]=mio(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of a new dynamic formulation.
% It is assumed that the dynamics equations are in the followi
%       _             _
%      |  M   Psiq^T  | |  ddq    |       |   f    |
%      |              | |         | =
%      |  Psiq   0    | | lambda  |       | gamma  |
%       -             -
v=diag(M);
v=v.^-1;
Minv=diag(v);
Q=Psiq*Minv*Psiq';
MQ=pinv(Q,1.e-6);
ddqf=Minv*f;
H=Minv*Psiq'*MQ;
ddq=(ddqf-H*Psiq*ddqf)+H*gamma;
end % of function mio
```

```
function[ddq]=mioultra(M,Psiq,gamma,f)
% Purpose: Computation of accelerations in a multibody dynamic
%          by means of a new dynamic formulation.
% It is assumed that the dynamics equations are in the followi
%
%      _                _
%     |  M    Psiq^T    | |  ddq    |       |   f    |
%     |                 | |         | =
%     |  Psiq   0       | | lambda  |       | gamma  |
%      -                -
[m,n] = size(M);
Psiqp=pinv(Psiq,1.e-6);
R=M*Psiqp;
Id=eye(m);
E=(Id-Psiqp*Psiq);
Q=pinv(E*M*E,1.e-6);
ddq=Q*f+(Psiqp-Q*M*Psiqp)*gamma;
end % of function mioultra
```