Efficient Solving Methods Exploiting Sparsity of Matrix in Real-Time Multibody Dynamic Simulation with Relative Coordinate Formulation

Choi, Gyoojae*, Yoo, Yungmyun, Im, Jongsoon
Korea Automotive Technology Institute(KATECH), Chungnam 330-910, Korea

In this paper, new methods for efficiently solving linear acceleration equations of multibody dynamic simulation exploiting sparsity for real-time simulation are presented. The coefficient matrix of the equations tends to have a large number of zero entries according to the relative joint coordinate numbering. By adequate joint coordinate numbering, the matrix has minimum off-diagonal terms and a block pattern of non-zero entries and can be solved efficiently. The proposed methods, using sparse Cholesky method and recursive block mass matrix method, take advantages of both the special structure and the sparsity of the coefficient matrix to reduce computation time. The first method solves the $n \times n$ sparse coefficient matrix for the accelerations, where n denotes the number of relative coordinates. In the second method, for vehicle dynamic simulation, simple manipulations bring the original problem of dimension $n \times n$ to an equivalent problem of dimension 6×6 to be solved for the accelerations of a vehicle chassis. For vehicle dynamic simulation, the proposed solution methods are proved to be more efficient than the classical approaches using reduced Lagrangian multiplier method. With the methods computation time for real-time vehicle dynamic simulation can be reduced up to 14 per cent compared to the classical approach.

Key Words : Real-Time Dynamic Simulation, Multibody Dynamics, Exploiting Sparsity, Hardware-in-the-Loop (HIL) Simulation

1. Introduction

Real-time hardware-in-the-loop (HIL) simulation has been quite widely used in the automotive industry with the advent of complex electronic controls to reduce development time and cost (Besinger, 1995). To develop an accurate real-time HIL facility, a real-time vehicle dynamic analysis program with reasonable simulation accuracy is needed.

The computational speed of simulation strongly depends on the coordinate systems used to formulate the equations of motion and solution methods (Lee, 1998; Bae, 1999; Cuadrado, 1997). There are two extreme coordinate systems: the Cartesian and the relative joint coordinate systems. The Cartesian formulation uses a maximal set of absolute position and orientation coordinates for each body. It is convenient to use this formulation to represent a mechanical system. The major drawback of it is, due to the maximal setting, the loss of computational efficiency. In contrast to the formulation, relative coordinates on joints between bodies may be used to define positions and orientations of bodies relative to one another, yielding a minimal set of coordinates. The relative joint formulation has substantial analytical complexity, but computational efficiency is gained. Using these

* Corresponding Author,
E-mail : gjchoi@katech.re.kr
TEL : +82-41-559-3105; **FAX** : +82-41-559-3111
Korea Automotive Technology Institute(KATECH), 74 Yongjong-Ri, Pungse-Myon, Chonan-Si, Chungnam 330-910, South Korea.(Manuscript **Received** November 14, 2000; **Revised** May 17, 2001)

formulations, the equations of motion are derived in the form of linear equations about accelerations and Lagrangian multipliers. The accelerations are integrated to obtain velocities and positions in the next step. To reduce the matrix size of resultant linear equations, coordinate partitioning method (Amirouche, 1992; Wehage, 1982) and reduced Lagrangian multiplier method (Serban, 1997) are used. These methods transform the equations of motion to a minimal set of coordinates without exploiting the structure and sparsity of the matrix to minimize computation time, as is done in the relative joint formulation. In the Cartesian formulation, a topology based approach for exploiting sparsity of resultant block diagonal matrix is introduced by Serban (Serban, 1997).

This paper focuses on improving the efficiency of solving the system of linear equations that determines accelerations and Lagrangian multipliers in the relative joint formulation for real-time dynamic simulation. The coefficient matrix of linear equations has a particular structure in the relative joint formulation and a topology-induced sparsity pattern when solving for the unknowns.

In this paper, new methods for efficiently solving linear equations in multibody dynamics using the relative joint formulation for real-time simulation exploiting sparsity are presented. Sparse Cholesky method and recursive block mass matrix method are developed to solve the linear equations efficiently and reduce computation time further. The methods are applied to solve the equations of real-time vehicle dynamic simulation and the computation time is compared to that of the classical approach of reduced Lagrangian multiplier method.

2. Equation of Motion

The Kane's equations with undetermined multipliers for constrained multibody system derived by Wang (Wang, 1987) may be given in matrix form as

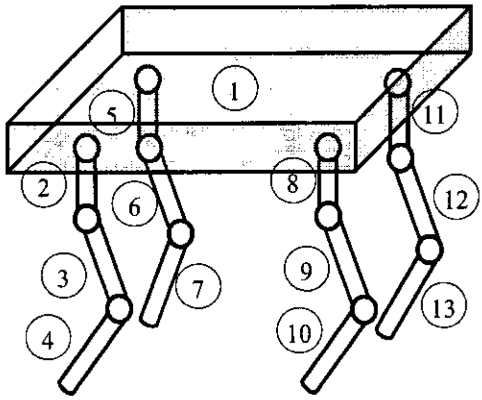


Fig. 1 Quadruped model

$$a\ddot{q}+b\dot{q}=f+B^T\lambda \tag{1}$$

$$\Phi(q)=0 \tag{2}$$

where q, $\dot{q}$ and $\ddot{q}$ are the vectors of dimension n representing the generalized position, velocity and acceleration of the system, respectively. In the equations f denotes the generalized external force vector, λ the Lagrangian multiplier vector of dimension m and $B=\frac{\partial\Phi}{\partial q}$ the $m\times n$ constraint Jacobian matrix, with $m<n$. The kinematic constraint equations $\Phi(q)$ are assumed to be linearly independent.

The constraint acceleration equations derived by successive time differentitations of Eq. (2) can be appended to the equations of motion in Eq. (1) to obtain a differential-algebraic equation as

$$\begin{pmatrix} a & B^T \\ B & 0 \end{pmatrix}\begin{Bmatrix} \ddot{q} \\ -\lambda \end{Bmatrix}=\begin{Bmatrix} f-b\dot{q} \\ \gamma \end{Bmatrix} \tag{3}$$

where the right side of the acceleration constraint equations is given as

$$\gamma=-(B\dot{q})_q\dot{q}-2B_t\dot{q}-\Phi_{tt} \tag{4}$$

In order for a simulation to progress to the next step, computation of the generalized accelerations $\ddot{q}$ from Eq. (3) and integrations are required.

3. Topology Based Sparsity Pattern Generation

The coefficient matrix of Eq. (3) tends to have a large number of zero entries and exploiting sparsity is necessary to solve the equation

Table 1 Bodies, joints and constraints of vehicle model

	No.	Description	No.	Description
Bodies	1	chassis	6	rear left knuckle
	2	rack	7	front right wheel
	3	front right knuckle	8	front left wheel
	4	front left knuckle	9	rear right wheel
	5	rear right knuckle	10	rear left wheel
	Abbr.	Description		
Joints and Constraints	SCJ1	front MacPherson strut suspension composite joint		
	SCJ2	rear MacPherson strut suspension composite joint		
	R	revolute joint		
	T	translational joint		
	DC	driving constraints		

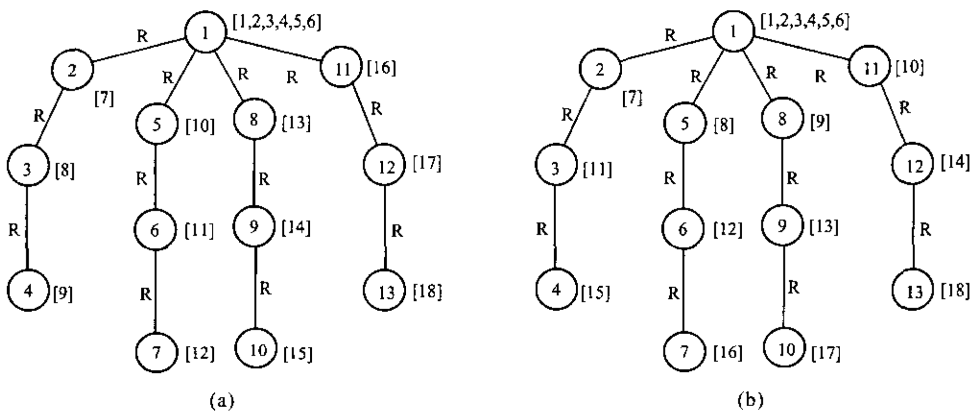
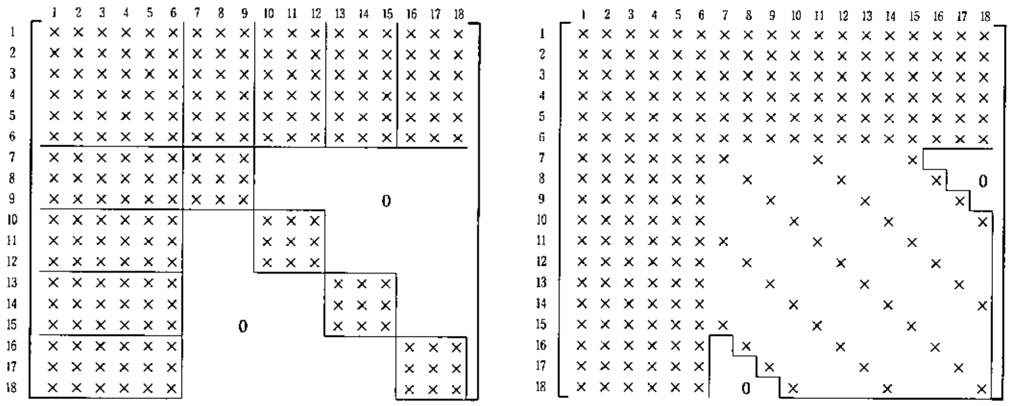


Fig. 2 Graphical representation of quadruped model

efficiently. The matrix a has the special pattern of sparsity called doubly bordered block diagonal (DBBD) matrix in relative joint fomulation. Two examples are used to illustrate this point. The first is the quadruped model presented in Fig. 1. The model is presented in Fig. 2 using graph and, in the figure, the bodies are represented as vertices and the joints as connecting edges. The numbers in the bracket mean the relative joint coordinate numbers. The two different generalized coordinate numbering sequences shown in Figs. 2(a) and 2(b) yield two different matrices shown in Figs. 3(a) and 3(b), respectively. In the figures, non-zero entries are denoted by x and there are many zero entries in the matrix. Figure. 3 shows that the pattern of matrix a and efficiency of a sparse matrix solver for it directly depend on the numbering scheme.

Selection of relative coordinate numbering is made in the modeling stage. For improved performance, it is important to determine the numbering sequence to minimize the bandwidth of off-diagonal entries, minimizing the amount of calculation in solving the matrix. The second example is a vehicle model presented in Fig. 4 with suspension composite joints and a compliant tie-rod model for real-time simulation. The joints and the model are used to reduce the computation time of simulation. One can refer to the paper by Choi (Choi, et. al., 2000) for the details of the composite joints and the model. Description of all the bodies and joints are listed in Table 1. For the model, two different generalized coordinate numbering sequences shown in Figs. 5(a) and 5 (b) yield two corresponding matrices a shown in Figs. 6(a) and 6(b). As seen in Figs. 3 and 6, relative coordinate numbering is critical in determining the sparsity pattern of matrix a. To



(a) (b)

Fig. 3 Pattern of the matrices with two relative joint numbering schemes of quadruped model (x: non-zero entries)

minimize the off-diagonal entries and to get the block pattern of the matrix, some numbering schemes has to be adapted. First, from the tree structure of the graph, the joint coordinates should be numbered sequentially along the body of each branch. Second, for a closed-loop mechanism, the cut joint position to make spanning tree may be located to minimize the number of joint coordinates at each branch and constraint equations. Using the proposed solving methods, the sparsity pattern of the matrix a will be used for efficient solving of the linear equations.

4. Solving Methods of the Equations of Motion

4.1 Reduced Langrangian multiplier method

Equation (1) is solved for $\ddot{q}$ in terms of unknown Lagrangian multiplier λ as

$$a\ddot{q}=(f-b\dot{q})+B^T\lambda \tag{5}$$

Assuming that a is nonsingular and using Eqs. (3) and (5), one can obtain the linear system

$$Ba^{-1}B^T\lambda=\gamma-Ba^{-1}(f-b\dot{q}) \tag{6}$$

This can then be solved for λ and, using Eq. (5), one can solve for generalized accelerations $\ddot{q}$. Thus, the maximum size of the coefficient matrix a of the equations is $n\times n$ and the sparsity pattern of matrix a can be used for advantage.

4.2 Sparse cholesky decomposition method

The pattern of matrix a is determined according to the relative coordinate numbering as shown in Figs. 3 and 6. The matrix a has the special form of sparse matrices as doubly bordered block diagonal (DBBD) matrix (Tewarson, 1972) in relative joint coordinate formulation when the sequence of row and column numbers are changed. In Cartesian formulation, it has the form of a banded diagonal matrix (Serban, 1997).

The solving algorithm for DBBD matrix are developed as shown in Fig. 7 using the Cholesky decomposition. Using the algorithm, one can efficiently solve for generalized acceleration by removing unnecessary zero (0) calculations. In Fig. 7, N is the size of matrix a, NQB the number of non-zero entities from diagonal element to backward, IDQB the column number according to NQB, NQF the number of non-zero entities from diagonal element to forward, and IDQF the column number according to NQF. The parameters are automatically decided from the topology information of the system. However, the Cholesky decomposition method has the disadvantage of requiring the evaluation of n square roots which, on a computer, usually takes much longer than other arithmetical operations (Jennings, 1977).

4.3 Recursive block mass matrix method

The coefficient matrix a shown in Fig. 6(a) can be partitioned with the non-zero block mass matrices and Eq. (5) may be rewritten as

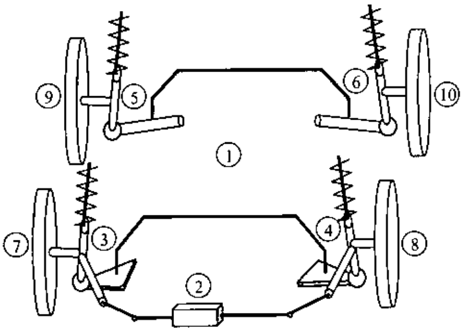


Fig. 4 Vehicle model

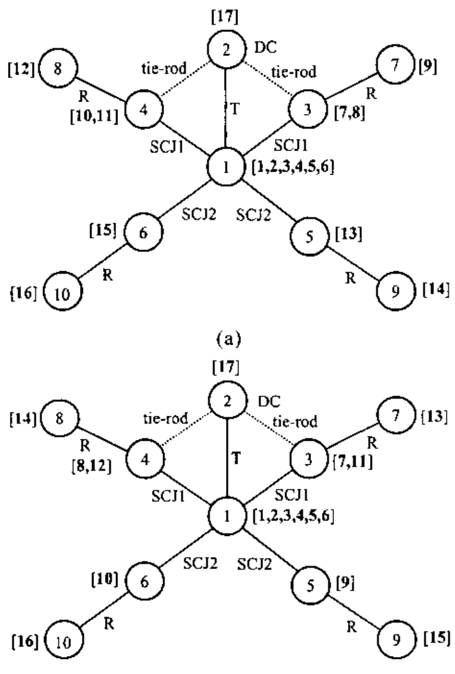


Fig. 5 Graphical representations of vehicle model

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} \\ A_{12}^T & A_{22} & 0 & 0 & 0 & 0 \\ A_{13}^T & 0 & A_{33} & 0 & 0 & 0 \\ A_{14}^T & 0 & 0 & A_{44} & 0 & 0 \\ A_{15}^T & 0 & 0 & 0 & A_{55} & 0 \\ A_{16}^T & 0 & 0 & 0 & 0 & A_{66} \end{bmatrix} \begin{Bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \\ \ddot{y}_3 \\ \ddot{y}_4 \\ \ddot{y}_5 \\ \ddot{y}_6 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{Bmatrix} \tag{7}$$

where, for a vehicle model, $\ddot{y}_1=(\ddot{q}_1\ddot{q}_2\ddot{q}_3\ddot{q}_4\ddot{q}_5\ddot{q}_6)^T$, $\ddot{y}_2=(\ddot{q}_7\ddot{q}_8\ddot{q}_9)^T$, $\ddot{y}_3=(\ddot{q}_{10}\ddot{q}_{11}\ddot{q}_{12})^T$, $\ddot{y}_4=(\ddot{q}_{13}\ddot{q}_{14})^T$, $\ddot{y}_5=(\ddot{q}_{15}\ddot{q}_{16})^T$ and $\ddot{y}_6=(\ddot{q}_{17})$ denote the relative coordinates of chassis, front right knuckle, front left knuckle, rear right knuckle, rear left knuckle

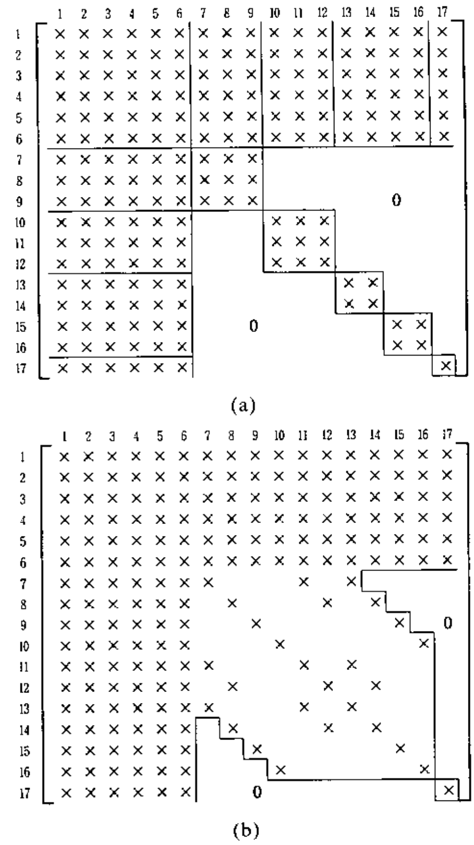


Fig. 6 Pattern of the matrices with two relative joint numbering schemes of vehicle model(x: non-zero entries)

and rack of a vehicle, respectively. And the right side of Eq. (5) is also partitioned for relative coordinates.

Eq. (7) can be written as

$$A_{11}\ddot{y}_1+\sum_{i=2}^{6}A_{1i}\ddot{y}_i=F_1 \tag{8}$$

$$A_{1i}^T\ddot{y}_1+A_{ii}\ddot{y}_i=F_i,\ i=2,\ \cdots\cdots,\ 6 \tag{9}$$

From Eq. (9) $\ddot{y}_i$, $i=2, \cdots\cdots, 6$ can be solved in terms of $\ddot{y}_1$ and substituted into Eq. (8) to derive the reduced equations of motion as

$$(A_{11}-\sum_{i=2}^{6}A_{1i}A_{ii}^{-1}A_{1i}^T)\ddot{y}_1=F_1-\sum_{i=2}^{6}A_{1i}A_{ii}^{-1}F_i \tag{10}$$

Eq. (10) is first solved to obtain the vector $\ddot{y}_1$ and then substituted into Eq. (9), to solve the acceleration vector $\ddot{y}_i$, $i=2, \cdots\cdots, 6$. This procedure can be used to solve Eq. (6) for Lagrangian multipliers. Thus the maximum matrix size to solve the equations with this process is 6×6 for vehicle dynamic analysis.

```
DOFOR I = 1,N
  DOFOR J = 1,NQB(I)
    JP = IDQB(J,I)
    SUM = A(I,JP)
    DOFOR K = 2,NQF(I)
      KP = IDQF(K,I)
        SUM = SUM-A(I,KP)*A(JP,KP)
    ENDDO
    IF (I=JP) THEN
      IF (SUM<=0) STOP 'FAILED'
        A(I,I) = SQRT(SUM)
    ELSE
      A(JP,I) = SUM/A(I,I)
    ENDIF
  ENDDO
ENDDO
```

(a) Decomposition

```
DOFOR I = 1,N
  SUM = B(I)
  DOFOR K = 2,NQF(I)
    KP = IDQF(K,I)
    SUM = SUM-A(I,KP)*X(KP)
  ENDDO
  X(I) = SUM/A(I,I)
ENDDO
DOFOR I = N,1,-1
  SUM = X(I)
  DOFOR K = 2,NQB(I)
    KP = IDQB(K,I)
    SUM = SUM-A(KP,I)*X(KP)
  ENDDO
  X(I) = SUM/A(I,I)
ENDDO
```

(b) Back substitution

Fig. 7 Modified Cholesky method for DBBD sparse matrix

5. Numerical Results and Discussion

To compare the performance of different solving methods, a vehicle model shown in Fig. 4 is used. The model is equipped with front and rear MacPherson strut suspension systems and these are modeled by front and rear MacPherson strut suspension composite joints (Choi, et. al., 2000). The model consists of 10 rigid bodies and has 16 degrees of freedom (17 relative coordinates, 1 driving constraint). All the three methods have the same computational work to generate Eq. (1). The proposed linear equation solvers are the sparse Cholesky method and recursive block mass matrix method. To solve Eqs. (5) and (6), the sparse Cholesky algorithm shown in Fig. 7 is used to remove unnecessary zero (0) calculations. For the recursive block mass matrix method, the maximum size of the coefficient matrix of Eq. (10) is 6×6.

All numerical experiments are performed on an ADRTS real-time computer with a PowerPC 604 333Mhz processor using non-optimized FORTRAN complier to compare the computation time clearly. Total computation times for a vehicle dynamic simulation are presented in Table 2. As seen in the table, the proposed two methods are more efficient than the classical one. These results show two ways of improving the efficiency of solving the linear equation. One is taking the advantage of exploiting sparsity and the other a better relative coordinate numbering within the matrix. The proposed solution techniques showed up to 14 per cent of total computation time reduction compared to that of the classical approach using reduced Lagrangian multiplier method in a vehicle model.

Table 2 Timing results for the vehicle model

Linear solving method	Computation time [seconds]	Speed up ratio [%]
Reduced Lagrangian multiplier method	10.42	100
Sparse Cholesky method	9.84	94.4
Recursive block mass matrix method	8.98	86.2

*) Simulation conditions;

1) Double lane change(ISO TR3888 severe lane change maneuver)
2) Simulation time : 0–10 seconds
3) Integration method : Euler method
4) Integration step size : 1. 2 milliseconds
5) Compiler : Non-optimized FORTRAN

6. Conclusion

New methods for efficient solving linear equations of multibody dynamic simulation in relative formulation for real-time simulation are presented. By adequate joint coordinate numbering, it is shown that one can obtain an efficient coefficient matrix which has minimal off-diagonal terms and block pattern of non-zero entries. A quadruped model and a vehicle model are presented to explain these properties. To solve the sparse coefficient matrix of linear equation, two methods are proposed. First, the sparse Cholesky method solves a sparse $n \times n$ coefficient matrix for accelerations, where n denotes number of relative coordinates. Second, using recursive block mass matrix method for vehicle dynamic simulation, simple manipulations transform the original problem of dimension $n \times n$ to an equivalent problem of dimension 6×6 to be solved for the accelerations of a vehicle chassis. The proposed solution techniques proved to be more efficient than the classical approach up to 14 per cent of total computation time in a vehicle model.

References

Amirouche, F. M. L, 1992, *Computational Methods in Multibody Dynamics*, Prentice-Hall International, Inc.

Bae, D. S., Lee, J. K. Cho, H. J. and Yae, H., 2000, "An Explicit Integration Method for Realtime Simulation of Multibody Vehicle Models," *Computational Methods in Applied Mechanics and Engineering*, Vol. 187, pp. 337~350.

Besinger, F. H., Cebon, D. and Cole, D. J., 1995, "Force Control of a Semi-Active Damper," *Vehicle System Dynamics*, Vol. 24, pp. 695~723.

Choi, G. J., Yoo, Y. M., Lee, K. P., Yoon, Y. S., 2000, "A Real-Time Multibody Vehicle Dynamic Analysis Method Using Suspension Composite Joints," *International Journal of Vehicle Design*, Vol. 24, pp. 259~273.

Cuadrado, J., Cardenal, J. and Bayo, E., 1997, "Modeling and Solution Methods for Efficient Real-Time Simulation of Multibody Dynamics," *Multibody System Dynamics*, Vol. 1, pp. 259~280.

Jennings, A., 1977, *Matrix Computation for Engineers and Scientists*, John Wiley & Sons.

Lee, K. P. and Yoon, Y. S., 1998, "Efficiency Comparision of Multibody Dynamic Analysis Algorithms for Real-Time Simulation," *JSME International Journal*, Series C, Vol. 41, pp. 813~821.

Serban, R., Negrut, D., Haug, E. J. and Potra, F. A., 1997, "A Topology-Based Approach for Exploiting Sparsity in Multibody Dynamics in Cartesian Formulation," *Mechanics of Structures and Machines*, Vol. 25, pp. 379~396.

Tewarson, R. P., 1972, *Sparse Matrices*, New York : Academic Press.

Wang, J. T. and Huston, R. L., 1987, "Kane's Equations with Undetermined Multipliers-Approach to Constrained Multibody Systems," *ASME Journal of Applied Mechanics*, Vol. 54, pp. 424~429.

Wehage, R. A. and Haug, E. J., 1982, "Generalized Coordinate Partitioning of Dimension Reduction in Analysis of Constrained Dynamic Systems," *ASME Journal of Mechanical Design*, Vol. 104, pp. 247~255.