Subsystem Synthesis Methods with Independent Coordinates for Real-Time Multibody Dynamics

Sung-Soo Kim*
Department of Mechatronics Engineering, Chungnam National University, 220 Kung-dong, Yusong-ku, Daejeon 305-764, Korea
Ji-Hyeun Wang
Agency for Defense Development, Yusong P.O.Box 35-1, Daejeon 305-600, Korea

For real time dynamic simulation, two different subsystem synthesis methods with independent generalized coordinates have been developed and compared. In each formulation, the subsystem equations of motion are generated in terms of independent generalized coordinates. The first formulation is based on the relative Cartesian coordinates with respect to moving subsystem base body. The second formulation is based on the relative joint coordinates using recursive formulation. Computational efficiency of the formulations has been compared theoretically by the arithmetic operational counts. In order to verify real-time capability of the formulations, bump run simulations of a quarter car model with SLA suspension subsystem have been carried out to measure the actual CPU time.

Key Words : Realtime Dynamic Simulation, Subsystem Synthesis Method

1. Introduction

Realtime dynamic formulations for multibody systems are essential for hardware-in-the loop simulations or operator-in-the loop simulations. Especially, for multibody systems with identical independent subsystems, the subsystem synthesis method has been proposed (Kim, 2002). In the method, equations of motion for subsystems and the base body equations of motion are formed separately for efficient computation. For example, in the case of full vehicle model with 4 independent suspension subsystems, the method has been proved to be more than 4 times faster than the conventional method, since the method deals with several smaller size equations of motion (Oh and Kim, 2001).

There are two distinct coordinate systems normally used in the multibody dynamics formulations ; one is Cartesian coordinate system and the other is relative joint coordinate system. In the Cartesian coordinate formulations, equations of motions are formed for each body, and constraint equations are systematically imposed for connections of bodies. Consequently, these formulations lead to general algorithms with easy implementation. However, they must deal with large size of DAE (Differential Algebraic Equations). Although these kinds of formulations are not efficient for serial computations, they have potential for body by body parallel computation. On the other hand, in the relative coordinate formulations, topological information of the system must be used. Velocity transformation from the Cartesian to relative joint velocities is applied recursively along branches of the topological tree

* Corresponding Author,
E-mail : sookim@cnu.ac.kr
TEL : +82-42-821-6872; **FAX** : +82-42-823-4919
Department of Mechatronics Engineering, Chungnam National University, 220 Kung-dong, Yusong-ku, Daejeon 305-764, Korea. (Manuscript **Received** November 29, 2004; **Revised** December 15, 2004)

structures (Tsai and Haug, 1989). These formulations are difficult to generalize. However, they are efficient, because they deal with the minimum number of generalized coordinates and constraint equations.

In this paper, two different subsystem synthesis methods with independent generalized coordinates are developed and compared to investigate the effectiveness and efficiency for the realtime simulations of multibody systems. As for the first formulation, a relative Cartesian coordinate formulation is proposed for subsystem equations of motion. To increase computational efficiency, additional independent coordinates are introduced and the subsystem equations of motion in terms of relative Cartesian coordinates are transformed into those in terms of the introduced independent coordinates. The reason to investigate the relative Cartesian coordinate formulation in this research is that the formulation has implementation easiness and has potential for parallel computations. As for the second formulation, the subsystem synthesis method with relative joint coordinates is investigated. Using generalized coordinate partitioning method, subsystem equations of motion are formed in terms of independent joint coordinates.

2. Overview of the Subsystem Synthesis Method

A typical multibody model with n independent subsystems is considered as shown in Fig. 1. A subsystem consists of bodies, joints, and force elements, and virtual base body which is the reference body to define kinematic relationship between bodies in the subsystem.

In the subsystem synthesis method, effective mass matrix and force vector must be computed first in each subsystem. The effective mass and force, in fact, represent the dynamic effects due to physical coupling between the subsystem and the base body. With the effective mass matrices and force vectors from each of the subsystems, the following 6×6 matrix form of equations of motion for the base body is obtained to solve for accelerations.

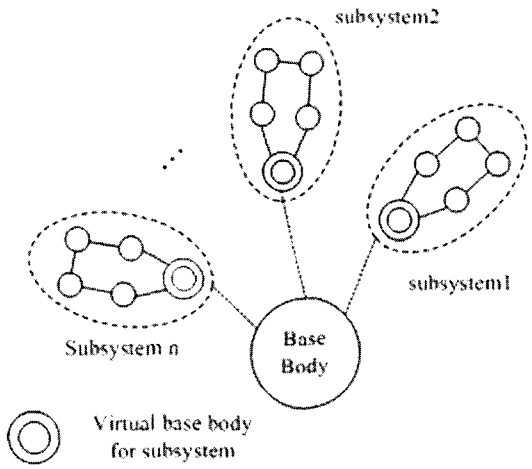

Fig. 1 Subsystems topology

$$\left(\hat{\mathbf{M}}_0+\sum_{i=1}^{n}\breve{\mathbf{M}}_i\right)\dot{\hat{\mathbf{Y}}}=\left(\hat{\mathbf{Q}}_0+\sum_{i=1}^{n}\breve{\mathbf{P}}_i\right) \tag{1}$$

where $\breve{\mathbf{M}}_i$ and $\breve{\mathbf{P}}_i$ (i=1~n) are the effective inertia matrix and the effective force vector of each subsystem, respectively, and $\dot{\hat{\mathbf{Y}}}$ is the base body acceleration, $\hat{\mathbf{M}}_0$ and $\hat{\mathbf{Q}}_0$ are the inertia matrix and generalized force vector of the base body, respectively. After solving for the base body acceleration, the following subsystem equations of motion can be solved for subsystem accelerations.

$$\begin{bmatrix}\mathbf{M}_{qq} & \boldsymbol{\Phi}_{\bar{q}}^{T}\\ \boldsymbol{\Phi}_{\bar{q}} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\ddot{\mathbf{q}}\\ \boldsymbol{\lambda}\end{bmatrix}=\begin{bmatrix}\mathbf{P}_q\\ \boldsymbol{\gamma}\end{bmatrix}-\begin{bmatrix}\mathbf{M}_{yq}^{T}\\ \mathbf{0}\end{bmatrix}\dot{\hat{\mathbf{Y}}}_0 \tag{2}$$

where $\ddot{\mathbf{q}}$ is the acceleration vectors represented in the coordinate system used in the subsystems, $\mathbf{M}_{qq}$, $\mathbf{M}_{yq}^{T}$ and $\mathbf{P}_q$ subsystem matrix and force vector, respectively. $\boldsymbol{\Phi}_{\bar{q}}$ is the constraint Jacobian matrix in the subsystem, $\boldsymbol{\lambda}$ is the Lagrange multiplier and $\boldsymbol{\gamma}$ is the right hand side of the constraint acceleration equations.

The subsystem synthesis method naturally provides modular structure. Thus, as long as the effective mass matrices and force vectors are generated, different multibody dynamics formulations of a subsystem module can be easily implemented. In following sections, two different formulations with independent coordinates are presented for the subsystem synthesis method to investigate their effectiveness and efficiency. One

is based on relative Cartesian coordinates and the other is based on relative joint coordinates.

3. Relative Cartesian Coordinate Formulation

3.1 Kinematics in terms of relative Cartesian coordinates

In order to derive subsystem equations of motion separately, motions of the bodies in the subsystem must be expressed with respect to the virtual reference body in Fig. 1. For this purpose, relative Cartesian coordinates are employed as shown in Fig. 2.

X-Y-Z reference frame represents the inertial reference frame, x′-y′-z′ frames represent the body fixed reference frame, and x_0^*-y_0^*-z_0^* frame represents the fixed reference frame of the body 0, which is moving with respect to the inertia reference frame. The orientation of the body i reference frame with respect to the inertial reference frame can be represented by the following sequence of coordinate transformation matrices;

$$\mathbf{A}_i=\mathbf{A}_o\mathbf{A}_{io} \tag{3}$$

where $\mathbf{A}_i$ is the orientation matrix of the body i, $\mathbf{A}_o$ is the orientation matrix of the virtual reference body 0, and $\mathbf{A}_{io}$ is the relative orientation of the body i with respect to body 0. The position vector $\mathbf{r}_i$ of the body i can be written as;

$$\mathbf{r}_i=\mathbf{r}_0+\mathbf{A}_0\mathbf{r}_{i0}^* \tag{4}$$

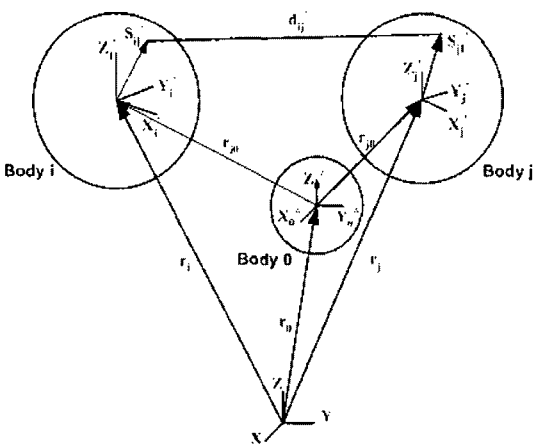

Fig. 2 Relative cartesian coordinate kinematics

where $\mathbf{r}_0$ is the position vector of the virtual reference body 0 and $\mathbf{r}_{i0}^*$ is the relative position vector of the body with respect to the body 0, and it is represented in x_0^*-y_0^*-z_0^* reference frame.

Angular velocity of the body i can be represented in terms of the angular velocity of the virtual reference body 0 and the relative angular velocity of body i with respect to body 0 as;

$$\boldsymbol{\omega}_i'=\mathbf{A}_{i0}^T\boldsymbol{\omega}_0^*\,\boldsymbol{\omega}_{i0}' \tag{5}$$

where $\boldsymbol{\omega}_i'$ is the angular velocity vector of the body i, represented in the body i reference frame, $\boldsymbol{\omega}_0^*$ is the angular velocity of the body 0, represented in body 0 reference frame, and $\boldsymbol{\omega}_{i0}'$ is the relative angular velocity of the body i with respect to body 0, represented in body i reference frame. Velocity of the body i is obtained by differentiating Eq. (5) with respect to time as;

$$\dot{\mathbf{r}}_i=\dot{\mathbf{r}}_0+\mathbf{A}_0\tilde{\boldsymbol{\omega}}_0^*\mathbf{r}_{i0}^* \tag{6}$$

where $\dot{\mathbf{r}}_i$ is the velocity of the body i, $\dot{\mathbf{r}}_0$ is the velocity of the virtual reference body 0, $\dot{\mathbf{r}}_{i0}^*$ is the relative Cartesian velocity vector of the body i with respect to body 0. In order to have more compact form of the equations, the composite velocity vector $\mathbf{y}_i=[\dot{\mathbf{r}}_i^T,\ \boldsymbol{\omega}_i^T]^T$, $\mathbf{y}_{i0}=[\dot{\mathbf{r}}_{i0}^{*T},\ \boldsymbol{\omega}_{i0}'^T]^T$, $\mathbf{y}_0=[\dot{\mathbf{r}}_0^T,\ \boldsymbol{\omega}_0^{*T}]^T$ are employed. Then, the following composite velocity relationship is obtained.

$$\mathbf{y}_i=\mathbf{E}_{i0}\mathbf{y}_0+\mathbf{G}_0\mathbf{y}_{i0} \tag{7}$$

where $\mathbf{E}_{i0}\equiv\begin{bmatrix}\mathbf{I} & -\mathbf{A}_0\tilde{\mathbf{r}}_{i0}^*\\ \mathbf{0} & \mathbf{A}_{i0}^T\end{bmatrix}$ and $\mathbf{G}_0\equiv\begin{bmatrix}\mathbf{A}_0 & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}$.

The acceleration relationship between body i and the virtual reference body 0 is obtained by differentiating Eq. (7) with respect to time as;

$$\dot{\mathbf{y}}_i=\mathbf{E}_{i0}\dot{\mathbf{y}}_0+\mathbf{G}_0\dot{\mathbf{y}}_{i0}+\mathbf{h}_{i0} \tag{8}$$

where $\mathbf{h}_{i0}=\begin{bmatrix}\mathbf{A}\tilde{\boldsymbol{\omega}}_0^*\tilde{\boldsymbol{\omega}}_0^*\mathbf{r}_{i0}^*+2\mathbf{A}_0\tilde{\boldsymbol{\omega}}_0^*\dot{\mathbf{r}}_{i0}^*\\ \tilde{\boldsymbol{\omega}}_i'\boldsymbol{\omega}_{i0}'\end{bmatrix}$ is the velocity coupling term.

3.2 Subsystem equations of motion using relative Cartesian coordinate formulation

Variational form of the subsystem equations of motion can be obtained by summation of the

virtual work form of the D'Alembert equations of the bodies in the subsystem.

$$\delta\bar{\mathbf{z}}^T\{\bar{\mathbf{M}},\ \dot{\bar{\mathbf{y}}}-\bar{\mathbf{g}}\}=0 \tag{9}$$

where $\delta\bar{\mathbf{z}}$ is composite virtual displacement and virtual rotation vector of all the bodies in a subsystem, $\dot{\bar{\mathbf{y}}}$ is the composite acceleration of all the bodies in a subsystem and $\bar{\mathbf{M}}$ is the block diagonal constant mass and inertia matrix of a subsystem, and $\bar{\mathbf{g}}$ is the generalized composite force vector acting on all the bodies in a subsystem.

The virtual displacement relationship between all the bodies and the reference body 0 in a subsystem can be obtained as

$$\delta\bar{\mathbf{z}}=\bar{\mathbf{E}}_{i0}\delta\mathbf{z}_0+\bar{\mathbf{G}}_0\delta\bar{\mathbf{z}}_{i0} \tag{10}$$

The acceleration relationship between all the bodies and the reference body 0 in a subsystem is obtained as ;

$$\dot{\bar{\mathbf{y}}}=\bar{\mathbf{E}}_{i0}\dot{\mathbf{y}}_0+\bar{\mathbf{G}}_0\dot{\bar{\mathbf{y}}}_{i0}+\bar{\mathbf{h}}_{i0} \tag{11}$$

In order to transform the equations of motion in Eq. (9) into the equations of motion in terms of relative Cartesian coordinates, substituting Eq. (10) and Eq. (11) into Eq. (9) yields following equations of motion ;

$$\begin{aligned}&\delta\mathbf{z}_0^T\{\bar{\mathbf{M}}_{EE}\dot{\mathbf{y}}_0+\bar{\mathbf{M}}_{EG}\dot{\bar{\mathbf{y}}}-\bar{\mathbf{g}}_E\}\\&+\delta\bar{\mathbf{z}}_{i0}^T\{\bar{\mathbf{M}}_{EG}^T\dot{\mathbf{y}}_0+\bar{\mathbf{M}}_{GG}\dot{\bar{\mathbf{y}}}_{i0}-\bar{\mathbf{g}}_G\}=0\end{aligned} \tag{12}$$

where, the virtual displacement $\delta\bar{\mathbf{z}}_{i0}$ is kinematically admissible. In the subsystem, bodies are connected with joints. Therefore, relative Cartesian coordinates must be satisfied constraint equations from kinematic joints in a subsystem as ;

$$\bar{\Phi}(\mathbf{r}_{10}^*\mathbf{A}_{10},\ \cdots,\ \mathbf{r}_{nb0}^*\mathbf{A}_{nb0})=0 \tag{13}$$

For real-time simulation, it is desirable to generate equations in terms of only independent generalized coordinates, because it produces ODE form of equations of motion. To select independent generalized coordinates, the generalized coordinate partitioning method (Haug, 1992) can be used as an application of the implicit function theorem. However, in the relative Cartesian coordinate system, a set of independent coordinates can have to be often changed during simulations, according to the configuration of the subsystem. As for the numerical analysis point of view, it is desirable to select different set of independent coordinates in the subsystem, which remain as independent coordinates in entire simulation. Thus, it is preferable to introduce new set of independent generalized coordinates as many as the same number of degrees of freedom (called *ndof*) in the subsystem. Then, there must be *ndof* constraint equations which make relationships between relative Cartesian coordinates and newly introduced independent generalized coordinates. Now, the constraint equations associated with the subsystem are as follows ;

$$\boldsymbol{\Psi}=\begin{bmatrix}\bar{\Phi}(\mathbf{r}_{10}^*\mathbf{A}_{10},\ \cdots,\ \mathbf{r}_{nb0}^*\mathbf{A}_{nb0})\\ \boldsymbol{\Gamma}(\mathbf{r}_{10}^*\mathbf{A}_{10},\ \cdots,\ \mathbf{r}_{nb0}^*\mathbf{A}_{nb0},\ \boldsymbol{\theta})\end{bmatrix}=\mathbf{0} \tag{14}$$

where $\boldsymbol{\Gamma}$ is the *ndof* constraint equations, and $\boldsymbol{\theta}$ is the new set of independent coordinate vector. Taking variation of the above constraint equations yields

$$\delta\boldsymbol{\Psi}=\boldsymbol{\Psi}_{\bar{z}_{i0}}\delta\bar{\mathbf{z}}_{i0}+\boldsymbol{\Psi}_{\theta}\delta\boldsymbol{\theta}=\mathbf{0} \tag{15}$$

where $\delta\boldsymbol{\theta}$ is the variation of the independent generalized coordinates. Constraint acceleration equations can be obtained by differentiating Eq. (14) twice with respect to time.

$$\boldsymbol{\Psi}_{\bar{z}_{i0}}\dot{\bar{\mathbf{y}}}_{i0}+\boldsymbol{\Psi}_{\theta}\ddot{\boldsymbol{\theta}}-\hat{\boldsymbol{\gamma}}=\mathbf{0} \tag{16}$$

where $\hat{\boldsymbol{\gamma}}=-(\dot{\boldsymbol{\Psi}}_{z_0}\bar{\mathbf{y}}_{i0}+\dot{\boldsymbol{\Psi}}_{\theta}\dot{\theta})$. From Eq. (15), virtual displacement vector of the relative Cartesian coordinate can be represented in terms of variation of the independent generalized coordinates as

$$\delta\bar{\mathbf{z}}_{i0}=-\boldsymbol{\Psi}_{\bar{z}_{i0}}^{-1}\boldsymbol{\Psi}_{\theta}\delta\boldsymbol{\theta}\equiv\mathbf{N}\delta\boldsymbol{\theta} \tag{17}$$

Similarly, acceleration in relative Cartesian coordinates can be expressed in terms of acceleration of the independent generalized coordinates as

$$\dot{\bar{\mathbf{y}}}_{i0}=-\boldsymbol{\Psi}_{\bar{z}_{i0}}^{-1}\boldsymbol{\Psi}_{\theta}\ddot{\boldsymbol{\theta}}+\boldsymbol{\Psi}_{\bar{z}_{i0}}^{-1}\hat{\boldsymbol{\gamma}}\equiv\mathbf{N}\ddot{\boldsymbol{\theta}}+\mathbf{p} \tag{18}$$

In order to derive subsystem equations of motion in terms of independent generalized coordinates, substituting Eq. (17) and Eq. (18) into Eq. (12) yields

$$\begin{aligned}&\delta\mathbf{z}_0^T\{\bar{\mathbf{M}}_{EE}\dot{\mathbf{y}}_0+\mathbf{M}_{E\theta}\ddot{\boldsymbol{\theta}}-\hat{\mathbf{g}}_E\}\\&+\delta\boldsymbol{\theta}^T\{\mathbf{M}_{E\theta}^T\dot{\mathbf{y}}_0+\mathbf{M}_{\theta\theta}\ddot{\boldsymbol{\theta}}-\mathbf{g}_{\theta}\}=0\end{aligned} \tag{19}$$

where $\mathbf{M}_{E\theta}=\bar{\mathbf{M}}_{EG}\mathbf{N}$, $\mathbf{M}_{\theta\theta}=\mathbf{N}^T\bar{\mathbf{M}}_{GG}\mathbf{N}$, $\hat{\mathbf{g}}_E=\bar{\mathbf{g}}_E-\bar{\mathbf{M}}_{EG}\mathbf{p}$, $\mathbf{g}_\theta=\mathbf{N}^T(\bar{\mathbf{g}}_G-\bar{\mathbf{M}}_{GG}\mathbf{p})$. Since variations of independent generalized coordinates are arbitrary, subsystem equations of motion is obtained as

$$\ddot{\boldsymbol{\theta}}=\mathbf{M}_{\theta\theta}^{-1}(\mathbf{g}_\theta-\mathbf{M}_{E\theta}^T\ddot{\mathbf{y}}_0) \tag{20}$$

In order to complete subsystem synthesis method using independent generalized coordinate formulation, effective mass matrix and force vectors described in Section 2 can be obtained by substituting Eq. (20) into Eq. (19) as,

$$\breve{\mathbf{M}}^c=\bar{\mathbf{M}}_{EE}-\mathbf{M}_{E\theta}\mathbf{M}_{\theta\theta}^{-1}\mathbf{M}_{E\theta}^T \tag{21}$$

$$\breve{\mathbf{g}}^c=\hat{\mathbf{g}}_E-\mathbf{M}_{E\theta}\mathbf{M}_{\theta\theta}^{-1}\mathbf{g}_\theta \tag{22}$$

It is noted that the dimension of matrix $\mathbf{M}_{\theta\theta}$ is $ndof\times ndof$. If the number of degrees of freedom in the subsystem is one or two, the inversion of matrix $\mathbf{M}_{\theta\theta}$ is not expensive. Once acceleration of the independent generalized coordinates is obtained, then the relative Cartesian coordinates in the subsystem can computed using Eq. (18).

4. Relative Joint Coordinate Formulations

4.1 Kinematics in terms of relative joint coordinate

To define kinematics in terms of relative joint coordinates in the subsystem, a pair of rigid bodies is shown in Fig. 3.

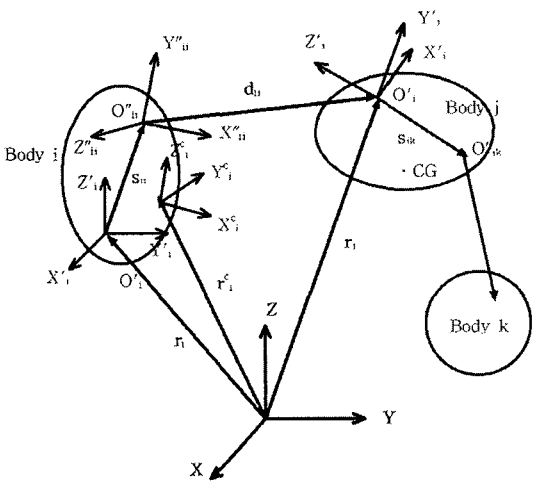

Fig. 3 A pair of connected bodies

The orientation of the body j reference frame with respect to the inertial reference frame can be represented by the following sequence of direction cosine matrices.

$$\mathbf{A}_j=\mathbf{A}_i\mathbf{C}_{ij}\mathbf{A}''_{ij} \tag{23}$$

where $\mathbf{A}_i$ and $\mathbf{A}_j$ are the orientation matrices of the body i and j, respectively, $\mathbf{C}_{ij}$ is the orthogonal transformation matrix from the $x''_{ij}-y''_{ij}-z''_{ij}$ frame to the $x'_i-y'_i-z'_i$ frame. $\mathbf{A}''_{ij}$ is the orthogonal transformation matrix from the $x'_j-y'_j-z'_j$ frame to the $x''_{ij}-y''_{ij}-z''_{ij}$ frame. The position vector $\mathbf{r}_j$ of the body j can be written as;

$$\begin{aligned}\mathbf{r}_j&=\mathbf{r}_i+\mathbf{s}_{ij}+\mathbf{d}_{ij}\\&=\mathbf{r}_i+\mathbf{A}_i\mathbf{s}'_{ij}+\mathbf{A}_i\mathbf{C}_{ij}\mathbf{d}''_{ij}(\mathbf{q}_j)\end{aligned} \tag{24}$$

where $\mathbf{r}_j$ is the position vector of the body j, and the position vector $\mathbf{s}_{ij}$ is the vector from the origin of the body i reference frame $\mathbf{O}'_i$ to that of the joint reference frame $\mathbf{O}''_{ij}$, and the position vector $\mathbf{d}_{ij}$ is the vector from the origin of the joint reference frame $\mathbf{O}''_{ij}$ to that of the body j reference frame $\mathbf{O}'_j$. It is noted that $\mathbf{d}_{ij}$ is a function of relative joint coordinate $\mathbf{q}_j$.

The compact form of velocity relationships between two bodies can be obtained, using the state vector notation as (Tsai and Haug, 1989);

$$\hat{\mathbf{Y}}_j=\hat{\mathbf{Y}}_i+\mathbf{B}_j\dot{\mathbf{q}}_j \tag{25}$$

where $\hat{\mathbf{Y}}_j$ and $\hat{\mathbf{Y}}_i$ are the state vectors of bodies i and j, respectively, and $\mathbf{B}_j$ is the velocity transformation matrix from the joint space to the state vector space. The acceleration relationship between two bodies is obtained by differentiating Eq. (24) as;

$$\dot{\hat{\mathbf{Y}}}_j=\dot{\hat{\mathbf{Y}}}_i+\mathbf{B}_j\ddot{\mathbf{q}}_j+\dot{\mathbf{B}}_j\dot{\mathbf{q}}_j=\dot{\hat{\mathbf{Y}}}_i+\mathbf{B}_j\ddot{\mathbf{q}}_j+\mathbf{D}_j \tag{26}$$

Position, velocity, and acceleration relationship can be applied recursively from the virtual reference body to the tree end bodies, if spanning tree structure is formed by virtual cut of some joints in the subsystem.

4.2 Subsystem equations of motion using relative joint coordinate formulation

Different from the relative Cartesian coordinate formulation, in the relative joint coordinate formulation, only cut joint constraints are

considered in the subsystem. If the virtual cut has been made between body n and n', the cut joint constraint equations can be expressed as a function of positions and orientations of body n and n' as ;

$$\boldsymbol{\Phi}(\mathbf{r}_n, \mathbf{A}_n, \mathbf{r}_{n'}, \mathbf{A}_{n'}) = \mathbf{0} \tag{27}$$

Using the Lagrange multiplier theorem (Haug, 1992), the variational equations of motion for the closed loop subsystem are obtained in the state vector space, as follows ;

$$\begin{aligned} &\delta\hat{\mathbf{Z}}_0^T(\hat{\mathbf{M}}_0\dot{\hat{\mathbf{Y}}}-\hat{\mathbf{Q}}_0)+\sum_{i=1}^{n}\delta\hat{\mathbf{Z}}_i^T(\hat{\mathbf{M}}_i\dot{\hat{\mathbf{Y}}}_i-\hat{\mathbf{Q}}_i) \\ &+\delta\hat{\mathbf{Z}}_n^T\boldsymbol{\Phi}_{\hat{Z}_n}^T\boldsymbol{\lambda}+\sum_{i=1'}^{n'}\delta\hat{\mathbf{Z}}_i^T(\hat{\mathbf{M}}_i\dot{\hat{\mathbf{Y}}}-\hat{\mathbf{Q}}_i) \\ &+\delta\hat{\mathbf{Z}}_{n'}^T\boldsymbol{\Phi}_{\hat{Z}_{n'}}^T\boldsymbol{\lambda}=0 \end{aligned} \tag{28}$$

where $\delta\hat{\mathbf{Z}}_i$ is the kinematically admissible generalized virtual displacements associated with the state vector. $\hat{\mathbf{M}}_i$ and $\hat{\mathbf{Q}}_i$ are the state mass matrix and the state generalized force vector, respectively. The first term of the Eq. (28) represents virtual work form of equations of motion for the subsystem virtual base body 0, the second and the fourth terms represent the equations of motion of the branches in the spanning tree of the subsystem, and finally the third and fifth terms are virtual work due to the constraint reaction forces that are acting on body n and n'. If the recursive formulation in the reference (Haug, 1992) is applied to branches of spanning tree and constraint acceleration equations in terms of the joint space is computed, the following subsystem equations of motion are obtained, in terms of relative joint coordinates and the base body coordinates.

$$\begin{bmatrix} \bar{\mathbf{M}}_{yy} & \bar{\mathbf{M}}_{yq} & \mathbf{0} \\ \bar{\mathbf{M}}_{yq}^T & \bar{\mathbf{M}}_{qq} & \boldsymbol{\Phi}_{\bar{\mathbf{q}}}^T \\ \mathbf{0} & \boldsymbol{\Phi}_{\bar{\mathbf{q}}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\hat{\mathbf{Y}}}_0 \\ \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{P}}_y \\ \bar{\mathbf{P}}_q \\ \boldsymbol{\gamma} \end{bmatrix} \tag{29}$$

By eliminating the first equation from Eq. (29), the subsystem equations of motion can be reduced down as ;

$$\begin{bmatrix} \bar{\mathbf{M}}_{qq} & \boldsymbol{\Phi}_{\bar{\mathbf{q}}}^T \\ \boldsymbol{\Phi}_{\bar{\mathbf{q}}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{P}}_q \\ \boldsymbol{\gamma} \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{M}}_{yq}^T \\ \mathbf{0} \end{bmatrix} \dot{\hat{\mathbf{Y}}}_0 \tag{30}$$

When the subsystem equations of motion are expressed in terms of the relative joint coordinates and unknown Lagrange multiplier as shown in Eq. (30), the DAE form of the equations of motion can be transformed into the ODE form of the equations of motion. Different from the case of the relative Cartesian coordinate formulation, the Generalized Coordinate Partitioning(GCP) technique can be used to select the independent coordinates $\mathbf{v}$ from the relative joint coordinate $\mathbf{q}$ in the subsystem, according to the implicit function theorem (Huag, 1992). Then, the equations of motion in terms of independent coordinates $\mathbf{v}$ can be obtained as ;

$$\mathbf{M}^*\dot{\mathbf{v}} = \mathbf{Q}_q^* - \mathbf{Q}_y^*\dot{\hat{\mathbf{Y}}}_0 \tag{31}$$

$$\ddot{\mathbf{u}} = \boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\gamma} - \boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_v\dot{\mathbf{v}} \tag{32}$$

where

$$\begin{aligned} \mathbf{M}^* = &\bar{\mathbf{M}}_{vv} - 2\bar{\mathbf{M}}_{vu}(\boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_{\bar{v}}) \\ &+ (\boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_{\bar{v}})^T\bar{\mathbf{M}}_{uu}(\boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_{\bar{v}}) \end{aligned} \tag{33}$$

$$\begin{aligned} \mathbf{Q}_q^* = &\bar{\mathbf{P}}_v - \bar{\mathbf{M}}_{vu}(\boldsymbol{\Phi}_{\bar{u}}^{-1}\bar{\boldsymbol{\gamma}}) - (\boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_{\bar{v}})^T\bar{\mathbf{P}}_u \\ &+ (\boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_{\bar{v}})^T\bar{\mathbf{M}}_{uu}(\boldsymbol{\Phi}_{\bar{u}}^{-1}\bar{\boldsymbol{\gamma}}) \end{aligned} \tag{34}$$

$$\mathbf{Q}_y^* = (\bar{\mathbf{M}}_{yq_u} - \bar{\mathbf{M}}_{yq_v}\boldsymbol{\Phi}_{\bar{u}}^{-1}\boldsymbol{\Phi}_{\bar{v}})^T \tag{35}$$

In order to complete the subsystem synthesis method using independent joint coordinate formulation, the effective mass matrix and force vector mentioned in Section 2 can be obtained by substituting Eq. (31) and (32) into the first equation of Eq. (29).

$$\breve{\mathbf{M}}^c = \bar{\mathbf{M}}_{yy} - \mathbf{Q}_y^{*T}(\mathbf{M}^*)^{-1}\mathbf{Q}_y^* \tag{36}$$

$$\breve{\mathbf{P}}^c = \bar{\mathbf{P}}_y - \bar{\mathbf{M}}_{yq_u}(\boldsymbol{\Phi}_{\bar{u}}^{-1}\bar{\boldsymbol{\gamma}}) - \mathbf{Q}_y^{*T}(\mathbf{M}^*)^{-1}\mathbf{Q}_q^* \tag{37}$$

After solving acceleration for independent coordinate $\dot{\mathbf{v}}$ in Eq. (31), dependent coordinate acceleration $\ddot{\mathbf{u}}$ can be computed using Eq. (32).

5. A Quarter Car Model with a SLA Suspension Subsystem

In order to investigate efficiency of three different formulations described in previous sections, a quarter car system with a SLA(short-long arm) suspension is modeled as shown in Fig. 4.

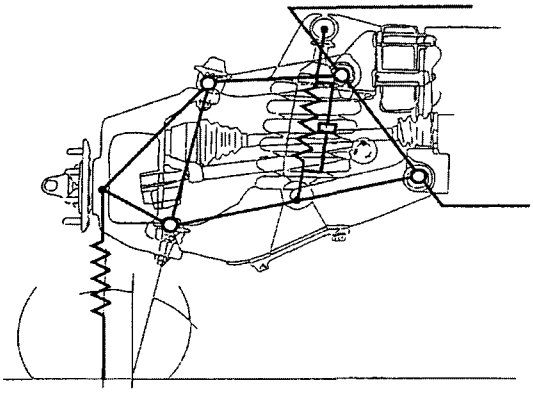

Fig. 4 A sla suspension subsystem

SLA suspension subsystem consist of a LCA (Lower control arm), a UCA (Upper Control Arm), a knuckle, and a tie rod. The LCA and the UCA are connected to the chassis with revolute joints, respectively. Spherical joints are used between the LCA and the knuckle and between the UCA and the knuckle. A tie rod connects the knuckle and the chassis. Since the mass and moment of inertia of the tie-rod are relatively small compared with other elements, the tie-rod is modeled as a mass-less link.

To compare the computational efficiency, arithmetic operators such as multiplication and addition are counted for the computations in the subsystem module, since other computations in different modules, such as chassis, tire, etc are the same.

Table 1 shows the comparison between two formulations for SLA suspension subsystem analysis based on an arithmetic operational counting method. M·D means number of multiplication or division and A·S denotes number of addition or subtraction operators. If the amount of CPU time spent for single operation of multiplication is the same as those for single operation of addition, then M·D and A·S can be added together to compare operational counts for the different formulations. Theoretical comparison shows that efficient formulation is the independent coordinate formulation with relative joint coordinates. Although the formulation based on relative joint coordinates is difficult to generalized, it is efficient.

Table 1 Theoretical comparison of two formulations

Com-putations	Formulations			
	Independent Generalized			
	with Relative Cartesian		with Relative Joint	
	M·D	A·S	M·D	A·S
Position	401	302	764	727
Velocity	414	302	290	220
RHS	245	196	127	194
Mass/Force	3,594	3,241	1,290	1,139
Effective mass/force	25	24	177	160
Acceleration	49	42	311	292
Subtotal	4,728	4,107	2,959	2,732
Total	8,835		5,691	
Ratio	1. 55		1	

In order to verify the realtime simulation capability of the model using subsystem synthesis method with independent coordinates, actual CPU times are measured for the bump run simulation. The bump run simulation of the quarter car model has been carried out using Fortran90 program in the PC with Intel Pentium-IV 1.6 Ghz CPU and 256 Mb Ram. The half sine bump with 0.5 m width and 0.05 m height was used. The initial velocity of vehicle is 10 km/h. Adams Bashforth 3rd order integration method with 8.3 ms step-size was used.

Fig. 5 shows the results of bump run simulations. There is small discrepancy. This might be due to different input data to describe the same suspension model. Essentially the identical results are obtained from two different formulations with independent coordinates. Table 2 shows CPU time comparison between two formulations. As shown in 3rd column of the table, both formulations provide realtime capability. The relative Cartesian coordinate formulation with independent coordinates is 3.27 times slower than the relative joint coordinate formulation for actual CPU time comparison, which is different from the theoretical investigation. Such discrepancy between actual CPU time measure and theoretical

Table 2 CPU time spent for bump run simulation

Independent coordinate Formulation	Average CPU Time spent (sec/frame)	Total CPU Time to Real Time (%)	Ratio of average CPU time of Relative Cartesian to Relative Joint
Relative Cartesian	0.00100	12.04	3.27
Relative Joint	0.000306	3.68	1

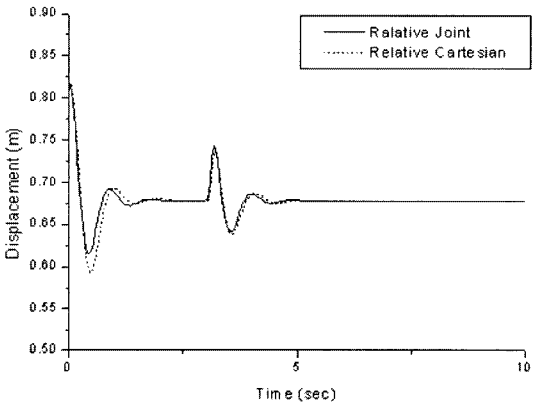

Fig. 5 Vertical displacements of c.g. of the chassis in bump run simulation

results is due to implementation of Fortan90 program, since the CPU time is also dependent on the number of argument variables in subroutine calls.

6. Conclusion

Two different formulations are investigated based on the subsystem synthesis method with independent generalized coordiantes, for realtime simulations of multibody systems. To compare the theoretical efficiency of the formulations, a quarter car model performance analysis has been carried out by counting arithmetic operators in the computations. Bump run simulations of the quarter car model have been carried out to measure actual CPU time. According to the CPU time results, The relative joint coordinate formulation based on indepedent coordinates is about 3.27 times faster than the relative Cartesian coordinates formualtion. However, the realtime simulation has been achieved with both formulations within about 12% of CPU time spent relative to the realtime. Thus, the relative Cartesian coordinate formulation with additional independent coordinates has also potential for realtime simulations and at the same time it provides generality and ease of implementation.

References

Haug, E. J., 1992, *Intermediate Dynamics*, Prentice Hall, Englewood Cliffs.

Kim Sung-Soo, 2002, "A Subsystem Synthesis Method for Efficient Vehicle Multibody Dynamics," *Multibody System Dynamics*, Vol. 7 No. 2, pp. 189~207.

Oh Young-Suk, Kim Sung-Soo, 2001, "A Real-Time Multibody Vehicle Dynamics Model Using a Subsystem Synthesis Method," *Proceedings of 2001 ASME. Design Engineering Technical Conference*, September 9-12.

Tsai, F. F. and Haug, E. J., 1989, "Automated Methods for High Speed Simulation of Multibody Dynamic Systems," Technical Report R-47, Center for Computer Aided Design, The University of Iowa.