Newton-Euler equations of multibody systems with changing structures for space applications

Vladimir Aslanov and Genrih Kruglov and Vadim Yudintsev

Korolev Samara State Aerospace University, Samara, Russia

Abstract

Multibody systems with changing structure are considered. These systems have stages in their motion that distinct from each other by degree of freedom (DOF), joint connection structure and joint types. These mechanical systems are common in space application e.g. separation subsystems. Single coordinate set is used to formulate Newton-Euler equations of motion at each stage. Proposed equation form simplifies equations building process for certain stages and whole motion. Numerical experiment was carried out using proposed method.

Key words: Multibody systems, Variable structure, Joint, Newton-Euler equation, Constraints, Rigid body

1 Introduction

Some aerospace systems belong to a system with changing structure [1], [2], [3], [4], [5]. It's possible to split their motion to stages or steps that distinct each other by the equation of motion. Transition from one stage to another occurs instantaneously. System structure, degrees of freedom, forces set are changed from stage to stage. Equations of motion of that complex multibody systems should be appropriate for building computational model. Many of the most efficient algorithms in multibody dynamics has developed over the last 30 years for robot applications [6], [7]. In [8] Featherstone reviews some of the accomplishments in the field of robot dynamics research (see also [9, chap. 5,6,7]). Santini and Gasbarri at [10] review some multibody dynamics algorithms for space applications.

Equations of motion can be written using two types of coordinates: relative coordinates that represent position and velocity of the body in terms of relative motion between interconnected bodies (joint coordinates) or absolute coordinates, i.e Cartesian. The choice of the coordinates is related to formulation of the dynamics of mechanical systems [11, chap. 1.2]. One of the effective method that uses joint coordinates is proposed in [12] by J. Wittenburg. Matrix equations are formed with no necessity to do any symbolical operations (e.g. symbolic differentiation). Minimal set of variables produce minimal set of equations without constraint forces. This method is appropriate for computer-aided analysis of mechanical systems. In the other hand we can choose absolute coordinates of bodies and write Newton-Euler equations for each body (see [11, chap. 1.2.3]), that leads to maximum number of equations. Obtained differential equation system must be solved with algebraic constraints equations.

Each of two methods can be used to build equations of motion for the considered mechanical systems. But we should take into account variations in system structure. Using first method we should build equations for each stage and coordinates set are different for each stage. It is necessary to do transform of coordinates during stage transition. These operations complicate our model. Second approach that used Newton-Euler equations allows to use the single coordinates set for all stages that simplify model building process. Structure change is described by including or excluding constraints equations. Constraints equations depend on joints that connect bodies. Newton-Euler equations with constraints equations forms DAE system [11]. Newton-Euler method leads to large set of equations but these equations have simple structure than equations obtained using relative (joint) coordinates. Also for open-loop systems several algorithm can be used to speed up calculation process (for example see Baraff [13] and Featherstone [14]). Our approach for mechanical systems with changing structures is based on Newton-Euler equations. Newton-Euler equations are solved with constraint equations that are built using two simple constraints. This two constraints allow simulate wide range of joints.

The rest of the paper is organized as follows: Section 2 defines the problem. In section 3 we describe two simple constraints and write down two constraints equations. In sections 4-5 we describe some numerical experiments to illustrate possibility of practical use of the offered equations form.

2 Statement of the problem

As we've noticed before Newton-Euler equation form is preferable for systems with changing structure: this equations are immutable from one motion phase to another and we need to change constraint equations only. DAE system for multibody dynamics can be written as [9, chap. 3.2]:

$$\begin{pmatrix} \mathbf{M} & \mathbf{Q}^T \\ \mathbf{Q} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \ddot{\mathbf{q}} \\ -\lambda \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{b} \end{pmatrix}, \tag{1}$$

here $\mathbf{q}$ is coordinates column matrix; $\mathbf{M}$ - is a generalized inertia matrix; $\mathbf{Q}$ - coefficients matrix of constraints equations, that can be written as:

$$\mathbf{Q}\ddot{\mathbf{q}} = \mathbf{b}.$$

Generalized inertia matrix $\mathbf{M}$ depends on $\mathbf{q}$. $\mathbf{Q}$ matrix depends on $\mathbf{q}$ and $\dot{\mathbf{q}}$. Complexity of these matrices depends on coordinates that used to describe bodies position and orientation. General inertia matrix has simplest structure for Newton-Euler equations (constant block-diagonal matrix). In this case, changing in the structure of the multibody system does not affect the structure of inertia matrix. Matrix $\mathbf{Q}$ depends of joints that connect bodies of mechanical systems [9]. The structure of this matrix and/or it size are changed with changes in system structure.

The method proposed here use two simple matrix constraint equations that enable us to write constraint equation for commonly used joints in mechanical systems for space application.

Newton-Euler equations for free rigid body are written as [15, chap. 7.11]:

$$\begin{pmatrix} \mathbf{m} & 0 \\ 0 & \mathbf{J}^{(c)} \end{pmatrix} \begin{pmatrix} \ddot{\mathbf{r}}^{(0)} \\ \dot{\omega}^{(c)} \end{pmatrix} = \begin{pmatrix} \mathbf{F}^{(0)} \\ \mathbf{L}^{(c)} \end{pmatrix} + \begin{pmatrix} 0 \\ -\tilde{\omega}^{(c)}\mathbf{J}^{(c)}\omega^{(c)} \end{pmatrix} \tag{2}$$

$\mathbf{m}$ - diagonal mass matrix; $\mathbf{J}^{(c)}$ - central inertia tensor; $\ddot{\mathbf{r}}^{(0)}$ - column of Cartesian coordinates relative to inertial frame; $\mathbf{F}^{(0)}$ - resultant force vector in inertial frame; $\mathbf{L}^{(c)}$ - resultant moment of the force system acting on the body; $\tilde{\omega}^{(c)}$ - angular velocity tensor:

$$\tilde{\omega}^{(c)} = \begin{pmatrix} 0 & -\omega_z^{(c)} & \omega_y^{(c)} \\ \omega_z^{(c)} & 0 & -\omega_x^{(c)} \\ -\omega_y^{(c)} & \omega_x^{(c)} & 0 \end{pmatrix}.$$

We will always use trailing superscript between parentheses to indicate frames in which vector or tensor coordinates are written.

For body in constrained multibody system reaction forces $\mathbf{R}^{(0)}$ and reaction torques must be added to right side of (2):

$$\begin{pmatrix} \mathbf{m} & 0 \\ 0 & \mathbf{J}^{(c)} \end{pmatrix} \begin{pmatrix} \ddot{\mathbf{r}}^{(0)} \\ \dot{\omega}^{(c)} \end{pmatrix} = \begin{pmatrix} \mathbf{F}^{(0)} + \mathbf{R}_{\lambda}^{(0)} \\ \mathbf{L}^{(c)} + \mathbf{L}_R^{(c)} \end{pmatrix} + \begin{pmatrix} 0 \\ \tilde{\omega}^{(c)}\mathbf{J}^{(c)}\omega^{(c)} \end{pmatrix} \tag{3}$$

Constraints equations should be solved with (3) to find reaction forces and torques $\mathbf{R}^{(0)}$, $\mathbf{L}_R^{(c)}$.

3 Constraint equations

Let us consider two simple constraints equations: "point on plane" and "relative rotation constraint". Constraint equations will be formulated in terms of accelerations for simultaneous solution of these equations with Newton-Euler equations.

3.1 Point on plane constraint

We suppose that contact point trajectory can be plane curve, straight line or contact point keeps position relative to ones body frame. This equation constraints relative linear motion of two bodies. *Contact point fixed on the one body* moves on plane that is fixed on other body.

Let us write "point on plane" constraint equation as scalar product of two vectors (fig. 1):

$$\vec{n}_i \cdot \vec{a}_i^r = 0, \tag{4}$$

$\vec{a}_i^r$– is a contact point acceleration relative to frame i that based on body i; $\vec{n}_i$ –

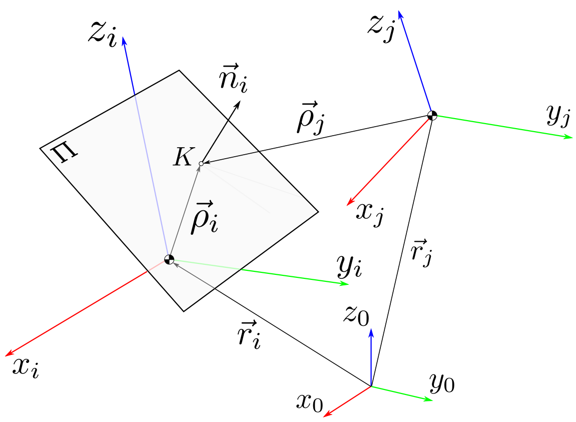

Figure 1. For "point on plane" equation

is a unit vector perpendicular to the plane based on body i. All time contact point must be on this plane. In matrix form equation (1) can be written as:

$$(\ddot{\rho}_i^{(i)})^T \mathbf{n}_i^{(i)} = 0, \tag{5}$$

$\ddot{\rho}_i^{(i)}$ - acceleration vector coordinates in the body i coordinate frame. This column matrix can be expressed using center of mass vectors. Firstly let us write $\rho_i^{(i)}$ as:

$$\rho_i^{(i)} = \mathbf{A}^{iT} \rho_i^{(0)}, \tag{6}$$

here $\rho_i^{(0)}$ – is a column vector $\vec{\rho}_i$ in the ground frame $O_o x_o y_o z_o$; $\mathbf{A}^i$ - transformation matrix of the vector coordinates in frame $O_i x_i y_i z_i$ into coordinates in frame $O_o x_o y_o z_o$. After differentiating (6) we have:

$$\dot{\rho}_i^{(i)} = \dot{\mathbf{A}}^{iT} \rho_i^{(0)} + \mathbf{A}^{iT} \dot{\rho}_i^{(0)}. \tag{7}$$

And taking into account that [12]:

$$\dot{\mathbf{A}}^{iT} = -\tilde{\omega}_i^{(i)} \mathbf{A}^{iT}, \ \dot{\mathbf{A}}^i = \mathbf{A}^i \tilde{\omega}_i^{(i)}, \tag{8}$$

we get:

$$\dot{\rho}_i^{(i)} = -\tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \rho_i^{(0)} + \mathbf{A}^{iT} \dot{\rho}_i^{(0)}. \tag{9}$$

Contact point vector $\vec{\rho}_i$ is expressed as sum of two center of mass vectors $\vec{r}_i, \vec{r}_j$ and contact point vector $\vec{\rho}_j$ (fig. 1):

$$\vec{\rho}_i = \vec{\rho}_j + \vec{r}_j - \vec{r}_i.$$

In matrix form:

$$\rho_i^{(0)} = \mathbf{A}^j \rho_j^{(j)} + \mathbf{r}_j^{(0)} - \mathbf{r}_i^{(0)}. \tag{10}$$

Taking into account (10) and assumption that column vector $\rho_j^{(j)}$ is constant in the frame $O_j x_j y_j z_j$ ($\rho_j^{(j)} = const$), contact point velocity relative to $O_i x_i y_i z_i$ in the frame $O_o x_o y_o z_o$ can be written as:

$$\dot{\rho}_i^{(0)} = \mathbf{A}^j \tilde{\omega}_j^{(j)} \rho_j^{(j)} + \dot{\mathbf{r}}_j^{(0)} - \dot{\mathbf{r}}_i^{(0)}. \tag{11}$$

After differentiating (9):

$$\ddot{\rho}_i^{(i)} = -\dot{\tilde{\omega}}_i^{(i)} \mathbf{A}^{iT} \rho_i^{(0)} + \tilde{\omega}_i^{(i)} \tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \rho_i^{(0)} - \\ - \tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \dot{\rho}_i^{(0)} - \tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \dot{\rho}_i^{(0)} + \mathbf{A}^{iT} \ddot{\rho}_i^{(0)}, \tag{12}$$

and now:

$$\ddot{\rho}_i^{(0)} = \dot{\mathbf{A}}^j \tilde{\omega}_j^{(j)} \rho_j^{(j)} + \mathbf{A}^j \dot{\tilde{\omega}}_j^{(j)} \rho_j^{(j)} + \ddot{\mathbf{r}}_j^{(0)} - \ddot{\mathbf{r}}_i^{(0)} = \\ \mathbf{A}^j \tilde{\omega}_j \tilde{\omega}_j^{(j)} \rho_j^{(j)} + \mathbf{A}^j \dot{\tilde{\omega}}_j^{(j)} \rho_j^{(j)} + \ddot{\mathbf{r}}_j^{(0)} - \ddot{\mathbf{r}}_i^{(0)}. \tag{13}$$

Now, let us substitute (13) into (12):

$$\ddot{\rho}_i^{(i)} = -\dot{\tilde{\omega}}_i^{(i)}\rho_i^{(i)} + \tilde{\omega}_i^{(i)}\tilde{\omega}_i^{(i)}\rho_i^{(i)} - \tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\dot{\rho}_i^{(0)} - \tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\dot{\rho}_i^{(0)} + \mathbf{A}^{iT}(\mathbf{A}^j\tilde{\omega}_j^{(j)}\tilde{\omega}_j^{(j)}\rho_j^{(j)} + \mathbf{A}^j\dot{\tilde{\omega}}_j^{(j)}\rho_j^{(j)} + \ddot{\mathbf{r}}_j^{(0)} - \ddot{\mathbf{r}}_i^{(0)}). \quad (14)$$

Rewrite last equation in order to group the terms containing accelerations:

$$\ddot{\rho}_i^{(i)} = \tilde{\rho}_i^{(i)}\dot{\omega}_i^{(i)} - \mathbf{A}^{iT}\mathbf{A}^j\tilde{\rho}_j^{(j)}\dot{\omega}_j^{(j)} + \mathbf{A}^{iT}\ddot{\mathbf{r}}_j^{(0)} - \mathbf{A}^{iT}\ddot{\mathbf{r}}_i^{(0)} + \tilde{\omega}_i^{(i)}\tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\rho_i^{(0)} - \tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\dot{\rho}_i^{(0)} - \tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\dot{\rho}_i^{(0)} + \mathbf{A}^{iT}\mathbf{A}^j\tilde{\omega}_j^{(j)}\tilde{\omega}_j^{(j)}\rho_j^{(j)} \quad (15)$$

Here the tilde operator transform any column vector $a = (a_x \ a_y \ a_z)^T$ into the skew-symmetric matrix:

$$\tilde{a} = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix}$$

After substituting (15) to (5) we have “point on plane” equation in form:

$$\mathbf{Q}_i\mathbf{X}_i + \mathbf{Q}_j\mathbf{X}_j = b_{ij}, \quad (16)$$

where the $\mathbf{Q}_i$ and $\mathbf{Q}_j$ are block matrices:

$$\mathbf{Q}_i = \left(-\mathbf{n}_i^{(i)T}\mathbf{A}^{iT}\middle|\mathbf{n}_i^{(i)T}\tilde{\rho}_i^{(i)}\right), \quad (17)$$

$$\mathbf{Q}_j = \left(\mathbf{n}_i^{(i)T}\mathbf{A}^{iT}\middle|-\mathbf{n}_i^{(i)T}\mathbf{A}^{iT}\mathbf{A}^j\tilde{\rho}_j^{(j)}\right). \quad (18)$$

$\mathbf{X}_i$, $\mathbf{X}_i$ are acceleration matrices:

$$\mathbf{X}_i = \begin{pmatrix} \ddot{\mathbf{r}}_i^{(0)} \\ \dot{\omega}_i^{(i)} \end{pmatrix}, \ \mathbf{X}_j = \begin{pmatrix} \ddot{\mathbf{r}}_j^{(0)} \\ \dot{\omega}_j^{(j)} \end{pmatrix};$$

and right side scalar:

$$b_{ij} = \mathbf{n}_i^{(i)T}(2\tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\dot{\rho}_i^{(0)} - \tilde{\omega}_i^{(i)}\tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\rho_i^{(0)} - \mathbf{A}^{iT}\mathbf{A}^j\tilde{\omega}_j^{(j)}\tilde{\omega}_j^{(j)}\rho_j^{(j)}). \quad (19)$$

The equation (16) is a scalar constraint equation. The reaction force f this constraint type is perpendicular to the plane Π. Let us assume that reaction force vector acts on body j in the direction of the vector $\vec{n}_i$, than reaction force vector that acts on body i in the opposite directions of the vector $\vec{n}_i$. We can write:

$$\mathbf{R}_j^{(0)} = -\mathbf{R}_i^{(0)} = \mathbf{n}_i^{(0)}\lambda = \mathbf{A}^i\mathbf{n}_i^{(i)}\lambda, \tag{20}$$

here λ is a Lagrange multiplier. This reaction force produces torque about the center of mass of the body j:

$$\mathbf{L}_{R_j}^{(j)} = \tilde{\rho}_j^{(j)}\mathbf{A}^{jT}\mathbf{A}^i\mathbf{n}_i^{(i)}\lambda. \tag{21}$$

Torque $\mathbf{L}_{R_j}^{(j)}$ is written in the body j frame. Coordinates of the torque vector that act on body i is written as:

$$\mathbf{L}_{R_i}^{(i)} = -\tilde{\rho}_i^{(i)}\mathbf{n}_i^{(i)}\lambda. \tag{22}$$

After making comparison between (17), (18) and (21), (22), reaction force and torque can be written as:

$$\begin{pmatrix} \mathbf{R}_j^{(0)} \\ \mathbf{L}_{R_j}^{(j)} \end{pmatrix} = \mathbf{Q}_j^T\lambda, \quad \begin{pmatrix} \mathbf{R}_i^{(0)} \\ \mathbf{L}_{R_i}^{(i)} \end{pmatrix} = \mathbf{Q}_i^T\lambda. \tag{23}$$

3.2 Relative rotation constraint

Here we derive constraint equation that restricts relative rotation of two bodies. This equation is formulated as follows: the projection of angular acceleration vector of one body relative to another body on the vector $\vec{n}_i$ must be zero:

$$\mathbf{n}_i^{(i)T}\varepsilon_{ij}^{(i)} = 0. \tag{24}$$

Relative angular velocity is defined as:

$$\omega_{ij}^{(i)} = \mathbf{A}^{iT}\mathbf{A}^j\omega_j^{(j)} - \omega_i^{(i)}. \tag{25}$$

After differentiate the (25) we get:

$$\varepsilon_{ij}^{(i)} = -\tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\mathbf{A}^j\omega_j^{(j)} + \mathbf{A}^{iT}\mathbf{A}^j\tilde{\omega}_j^{(j)}\omega_j^{(j)} + \mathbf{A}^{iT}\mathbf{A}^j\varepsilon_j^{(j)} - \varepsilon_i^{(i)}. \tag{26}$$

Substituting (26) into (24) leads to:

$$(\mathbf{A}^j\mathbf{A}^{iT}\mathbf{n}_i^{(i)})^T\varepsilon_j^{(j)} - \mathbf{n}_i^{(i)T}\varepsilon_i^{(i)} = \mathbf{n}_i^{(i)T}(\tilde{\omega}_i^{(i)}\mathbf{A}^{iT}\mathbf{A}^j\omega_j^{(j)} - \mathbf{A}^{iT}\mathbf{A}^j\tilde{\omega}_j^{(j)}\omega_j^{(j)}). \tag{27}$$

Equation (27) can be rewrite as follows:

$$\mathbf{Q}_i^r\mathbf{X}_i + \mathbf{Q}_j^r\mathbf{X}_j = b_{ij}^r, \tag{28}$$

Acceleration coefficient matrices (1×6) are written as:

$$
\begin{aligned}
\mathbf{Q}^r_j &= \left(\mathbf{0} \middle| -\mathbf{n}_i^{(i)T}\right), \\
\mathbf{Q}^r_i &= \left(\mathbf{0} \middle| (\mathbf{A}^j \mathbf{A}^{iT} \mathbf{n}_i^{(i)})^T\right),
\end{aligned}
\tag{29}
$$

b^r_{ij} is defined as:

$$
b^r_{ij} = \mathbf{n}_i^{(i)T} (\tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \mathbf{A}^j \omega_j^{(j)} - \mathbf{A}^{iT} \mathbf{A}^j \tilde{\omega}_j^{(j)} \omega_j^{(j)}). \tag{30}
$$

Reaction torque acting on body i:

$$
\mathbf{L}_i^{(i)} = (\mathbf{A}^j \mathbf{A}^{iT} \mathbf{n}_i^{(i)})^T \lambda, \tag{31}
$$

Reaction torque acting on body j

$$
\mathbf{L}_j^{(i)} = -\mathbf{n}_i^{(i)T} \lambda. \tag{32}
$$

For long time process we should take into account that equations (16) and (28) constraint only accelerations of bodies not position and velocity, than during numerical simulation the original constraint (5) may not fulfill and a drift-off from this constraint may occur. Several stabilization methods can be used to solve this problem [16], [17]. Using Baumgarte's method for "point on plane" constraints we should rewrite (38) as:

$$
\begin{aligned}
\mathbf{b}_{ij} = \mathbf{n}_i^{(i)T} (\tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \dot{\rho}_i^{(0)} - \tilde{\omega}_i^{(i)} \tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \rho_i \\
+ \tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \dot{\rho}_i^{(0)} - \mathbf{A}^{iT} \mathbf{A}^j \tilde{\omega}_j^{(j)} \tilde{\omega}_j^{(j)} \rho_j^{(j)}) + \beta^2 \epsilon_r + 2\alpha\epsilon_v,
\end{aligned}
\tag{33}
$$

here ϵ_r is a contact point position error – distance between contact point and the plane:

$$
\epsilon_r = \mathbf{n}_i^{(i)T} (\mathbf{A}^{iT} (\mathbf{A}^j \rho_j^{(j)} - \mathbf{r}_i^{(0)} + \mathbf{r}_j^{(0)}) - \mathbf{p}_i^{(i)}),
$$

$\mathbf{p}_i^{(i)}$ - is the coordinates of any point P on the plane Π in frame i (fig. 1). The ϵ_v is a contact point velocity error: the projection of the contact point velocity to the vector $\vec{n}_i^{(i)}$:

$$
\epsilon_v = \mathbf{n}_i^{(i)T} (-\tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \rho_i^{(0)} + \mathbf{A}^{iT} \dot{\rho}_i^{(0)}).
$$

Scalar parameters α and β should be [12]:

$$
\alpha > 0, \ \beta > 0.
$$

To apply Baumgarte's algorithm for the "relative rotation constraint" (30) should be rewritten as:

$$
b^r_{ij} = \mathbf{n}_i^{(i)T} (\tilde{\omega}_i^{(i)} \mathbf{A}^{iT} \mathbf{A}^j \omega_j^{(j)} - \mathbf{A}^{iT} \mathbf{A}^j \tilde{\omega}_j^{(j)} \omega_j^{(j)}) + \mathbf{n}_i^{(i)T} (\mathbf{A}^{iT} \mathbf{A}^j \omega_j^{(j)} - \omega_i^{(i)}) \sigma,
$$

where $\sigma > 0$.

Now using two types of constraint equations we can describe more complex joints. For example three "point on plane" constraint equations and two rotational constraint equations describe cylindrical joint with one rotational DOF. Below we consider some examples of using obtained constrain equations.

4 Three-link mechanism

At first let's consider a simple 3-link mechanism shown at fig 2 under the action of gravity force acting along the y_0 Cartesian direction. The rigid link 1 is connected to the ground and the link 2 by two spherical joints A and B respectively, the link 2 connected to link 3 by the cylindrical joint C. Each link has length $l_1 = l_2 = l_3 = l = 1\ m$, mass $m_1 = m_2 = m_3 = m = 1\ kg$ and inertia tensor in the principal axis:

$$\mathbf{J}_1^{(1)} = \mathbf{J}_2^{(2)} = \mathbf{J}_3^{(3)} = \begin{pmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{pmatrix}$$

where $Jx = 0.013\ kg{\cdot}m^2$ and $J_y = J_z = 0.083\ kg{\cdot}m^2$. At the joint A is defined

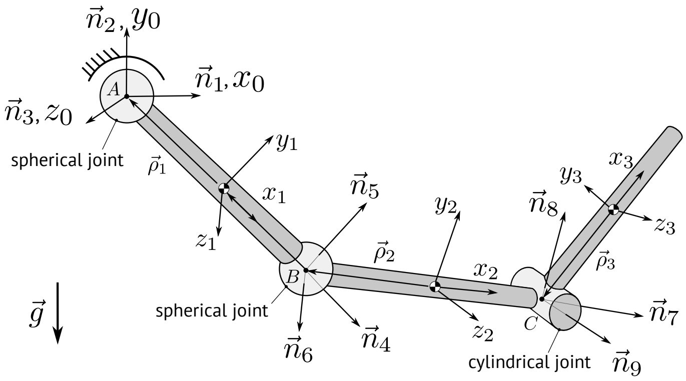

Figure 2. Three-link mechanism

three unit vectors $\vec{n}_1, \vec{n}_2, \vec{n}_3$ aligned with inertia axis x_0, y_0 and z_0 respectively. Point A and these vectors define three planes connected with ground. We can write three constraints equations using (16):

$$\mathbf{Q}_{0k}\mathbf{X}_0 + \mathbf{Q}_{1k}\mathbf{X}_1 = b_k, k = 1 \ldots 3 \tag{34}$$

We suppose that body 0 (ground) is fixed than $\ddot{\mathbf{X}}_0 = \mathbf{0}, \mathbf{A}^0 = \mathbf{E}_{3\times 3}$ (unit matrix), and last equation become the form:

$$\mathbf{Q}_{1k}\mathbf{X}_1 = b_k, k = 1 \ldots 3 \tag{35}$$

where the $\mathbf{Q}_{1k}$ is defined as:

$$\mathbf{Q}_{k1} = \left(\mathbf{n}_k^{(0)T} \middle| -\mathbf{n}_k^{(0)T}\mathbf{A}^1\tilde{\rho}_1^{(1)}\right),\ k = 1\dots3$$

and:

$$\mathbf{X}_1 = \left(\ddot{\mathbf{r}}_1^{(0)} \middle| \dot{\omega}_1^{(1)}\right)^T$$

The right side of the (35) is written as:

$$b_k = -\mathbf{A}^1\tilde{\omega}_1^{(1)}\tilde{\omega}_1^{(1)}\rho_1^{(1)},\ k = 1, 2, 3. \tag{36}$$

where

$$\rho_1^{(1)} = \begin{pmatrix} -\frac{l_1}{2} & 0 & 0 \end{pmatrix}^T = const.$$

Unit vectors coordinates in ground frame can be defined as:

$$\mathbf{n}_1^{(0)} = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T,\ \mathbf{n}_2^{(0)} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T,\ \mathbf{n}_3^{(0)} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T.$$

The constraint equations for joint B are written as:

$$\mathbf{Q}_{1k}\mathbf{X}_1 + \mathbf{Q}_{2k}\mathbf{X}_2 = b_k, k = 4\dots6 \tag{37}$$

where the $\mathbf{Q}_{1k}$ is defined as:

$$\mathbf{Q}_{1k} = \left(-\mathbf{n}_k^{(1)T}\mathbf{A}^{1T} \middle| \mathbf{n}_k^{(1)T}\tilde{\rho}_1^{(1)}\right),\ k = 4\dots6$$

and the row matrix $\mathbf{Q}_{2k}$:

$$\mathbf{Q}_{2k} = \left(\mathbf{n}_k^{(1)T}\mathbf{A}^{1T} \middle| -\mathbf{n}_k^{(1)T}\mathbf{A}^{1T}\mathbf{A}^2\tilde{\rho}_2^{(2)}\right),\ k = 4\dots6$$

Scalar b_k is written as:

$$\begin{aligned}\mathbf{b}_k = \mathbf{n}_k^{(1)T}(2\tilde{\omega}_1^{(1)}\mathbf{A}^{1T}\dot{\rho}_1^{(0)} - \tilde{\omega}_1^{(1)}\tilde{\omega}_1^{(1)}\mathbf{A}^{1T}\rho_1^{(0)} \\ - \mathbf{A}^{1T}\mathbf{A}^2\tilde{\omega}_2^{(2)}\tilde{\omega}_2^{(2)}\rho_2^{(2)}),\ k = 4, 5, 6.\end{aligned} \tag{38}$$

Unit vectors in the link 1 frame are defined as:

$$\mathbf{n}_4^{(1)} = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T,\ \mathbf{n}_5^{(1)} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T,\ \mathbf{n}_6^{(1)} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T.$$

These vectors define three planes connected with link 1.

To describe cylindrical joint C we should write three equations “point on plane“ and two equations that restrict rotation of the link 3 relative to the link 2 around axis defined by vectors $\vec{n}_{10}$ and $\vec{n}_{11}$. First three equations looks like constraints equations for joint B:

$$\mathbf{Q}_{2k}\mathbf{X}_2 + \mathbf{Q}_{3k}\mathbf{X}_3 = b_k, k = 7\dots 9 \tag{39}$$

where

$$\mathbf{Q}_{2k} = \left(-\mathbf{n}_k^{(2)T}\mathbf{A}^{2T}\middle|\mathbf{n}_k^{(2)T}\tilde{\rho}_2^{(2)}\right),\ k = 7\dots 9$$
$$\mathbf{Q}_{3k} = \left(\mathbf{n}_k^{(2)T}\middle|-\mathbf{n}_k^{(2)T}\mathbf{A}^3\tilde{\rho}_3^{(3)}\right),\ k = 7\dots 9$$

Unit vectors in the link 2 frame are defined as:

$$\mathbf{n}_7^{(2)} = \begin{pmatrix}1 & 0 & 0\end{pmatrix}^T,\ \mathbf{n}_8^{(2)} = \begin{pmatrix}0 & 1 & 0\end{pmatrix}^T,\ \mathbf{n}_9^{(2)} = \begin{pmatrix}0 & 0 & 1\end{pmatrix}^T.$$

Equations that restricts relative rotation:

$$\mathbf{Q}_{2k}\mathbf{X}_2 + \mathbf{Q}_{3k}\mathbf{X}_3 = b_k, k = 10, 11 \tag{40}$$

where

$$\mathbf{Q}_{2k} = \left(\mathbf{0}\middle|-\mathbf{n}_k^{(2)T}\right),\ k = 10, 11$$
$$\mathbf{Q}_{3k} = \left(\mathbf{0}\middle|(\mathbf{A}^{3T}\mathbf{A}^{2T}\mathbf{n}_k^{(2)})^T\right),\ k = 10, 11$$

Restrict rotation axes are defined as:

$$\mathbf{n}_{10}^{(2)} = \mathbf{n}_7^{(2)},\ \mathbf{n}_{11}^{(2)} = \mathbf{n}_8^{(2)}.$$

Let us write all equations that describe considered mechanism:

$$\begin{pmatrix}\mathbf{M}_1 & 0 & 0 & \mathbf{\Psi}_{11}^T & \mathbf{\Psi}_{12}^T & \mathbf{0}\\ 0 & \mathbf{M}_2 & 0 & \mathbf{0} & \mathbf{\Psi}_{22}^T & \mathbf{\Psi}_{23}^T\\ 0 & 0 & \mathbf{M}_3 & \mathbf{0} & \mathbf{0} & \mathbf{\Psi}_{33}^T\\ \mathbf{\Psi}_{11} & \mathbf{0} & \mathbf{0} & 0 & 0 & 0\\ \mathbf{\Psi}_{12} & \mathbf{\Psi}_{22} & \mathbf{0} & 0 & 0 & 0\\ \mathbf{0} & \mathbf{\Psi}_{23} & \mathbf{\Psi}_{33} & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}\mathbf{X}_1\\ \mathbf{X}_2\\ \mathbf{X}_3\\ \mathbf{\Lambda}_1\\ \mathbf{\Lambda}_2\\ \mathbf{\Lambda}_3\end{pmatrix} = \begin{pmatrix}\mathbf{P}_1\\ \mathbf{P}_2\\ \mathbf{P}_3\\ \mathbf{B}_1\\ \mathbf{B}_2\\ \mathbf{B}_3\end{pmatrix} \tag{41}$$

Block matrices $\mathbf{\Psi}_{1k}$ $(k = 1\dots 3)$ include matrices $\mathbf{Q}_{ij}$ pertaining to joint k:

$$\mathbf{\Psi}_{11} = \begin{pmatrix}\mathbf{Q}_{11}\\ \mathbf{Q}_{12}\\ \mathbf{Q}_{13}\end{pmatrix},\ \mathbf{\Psi}_{12} = \begin{pmatrix}\mathbf{Q}_{14}\\ \mathbf{Q}_{15}\\ \mathbf{Q}_{16}\end{pmatrix},\ \mathbf{\Psi}_{22} = \begin{pmatrix}\mathbf{Q}_{24}\\ \mathbf{Q}_{25}\\ \mathbf{Q}_{26}\end{pmatrix},$$

$$\mathbf{\Psi}_{23} = \begin{pmatrix} \mathbf{Q}_{27} \\ \mathbf{Q}_{28} \\ \mathbf{Q}_{29} \\ \mathbf{Q}_{2,10} \\ \mathbf{Q}_{2,11} \end{pmatrix}, \ \mathbf{\Psi}_{33} = \begin{pmatrix} \mathbf{Q}_{37} \\ \mathbf{Q}_{28} \\ \mathbf{Q}_{29} \\ \mathbf{Q}_{3,10} \\ \mathbf{Q}_{3,11} \end{pmatrix},$$

Column matrices $\mathbf{\Lambda}_k$ include Lagrange multipliers for the joint k:

$$\mathbf{\Lambda}_1 = \begin{pmatrix} \lambda_1 \ \lambda_2 \ \lambda_3 \end{pmatrix}, \ \mathbf{\Lambda}_2 = \begin{pmatrix} \lambda_4 \ \lambda_5 \ \lambda_6 \end{pmatrix}, \ \mathbf{\Lambda}_3 = \begin{pmatrix} \lambda_7 \ \lambda_8 \ \lambda_9 \ \lambda_{10} \ \lambda_{11} \end{pmatrix}$$

External forces and torques are contained in matrices $\mathbf{P}_i$:

$$\mathbf{P}_k = \begin{pmatrix} \mathbf{F}_k^{(0)} \\ \mathbf{L}_k^{(k)} + \tilde{\omega}_k^{(k)} \mathbf{J}_k^{(k)} \omega_k^{(k)} \end{pmatrix}, \ \mathbf{F}_k^{(0)} = \begin{pmatrix} 0 \ -mg \ 0 \end{pmatrix}, \ \mathbf{L}_k^{(k)} = \begin{pmatrix} 0 \ 0 \ 0 \end{pmatrix}.$$

Column matrices $\mathbf{B}_k$ include right side expressions of the constraint equations:

$$\mathbf{B}_1 = \begin{pmatrix} b_1 \big| b_2 \big| b_3 \end{pmatrix}^T, \ \mathbf{B}_2 = \begin{pmatrix} b_4 \big| b_5 \big| b_6 \end{pmatrix}^T, \ \mathbf{B}_3 = \begin{pmatrix} b_7 \big| b_8 \big| b_9 \big| b_{10} \big| b_{11} \end{pmatrix}^T$$

(41) should be supplemented by kinematic equations to obtain rotation matrices $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$, that can be expressed in terms of Euler angles, quaternions. For example, rotation matrices can be obtained from differential equations:

$$\dot{\mathbf{A}}^i = \mathbf{A}^i \tilde{\omega}_i^{(i)}. \tag{42}$$

The resulting equation set (41) with kinematic equations (42) was solved in MATLAB (Runge-Kutta rkf45 method is used). At fig. 3 several snapshots of the motion three-link mechanism is shown. Energy error for this conservative system is shown at fig 4. Energy error graph was found for two cases. In first case constraints equation are written without stabilization term. In second case constraints equations are written using Bumgarte's additional terms. By means of it energy error is reduced. Figure 5 illustrate position-level drift-off (ϵ_r) of the constraints 6 and 7 with and without Baumgarte's stabilization technique.

5 Separating first stage booster

Here we consider first stage separation subsystem of a "Soyuz"-like space carrier vehicle. During separation process either of the four first stage booster rotate about joint at them cone by the engine after-action pulse. First stage

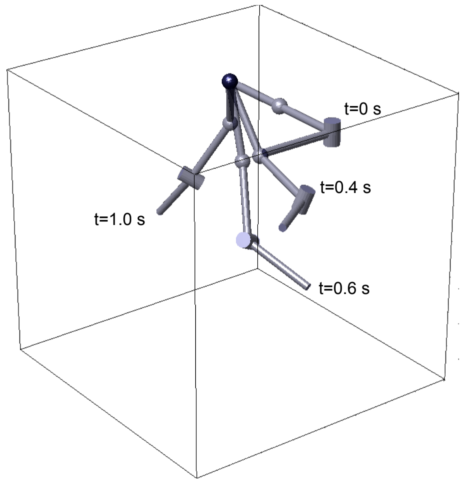

Figure 3. Three-link mechanism motion

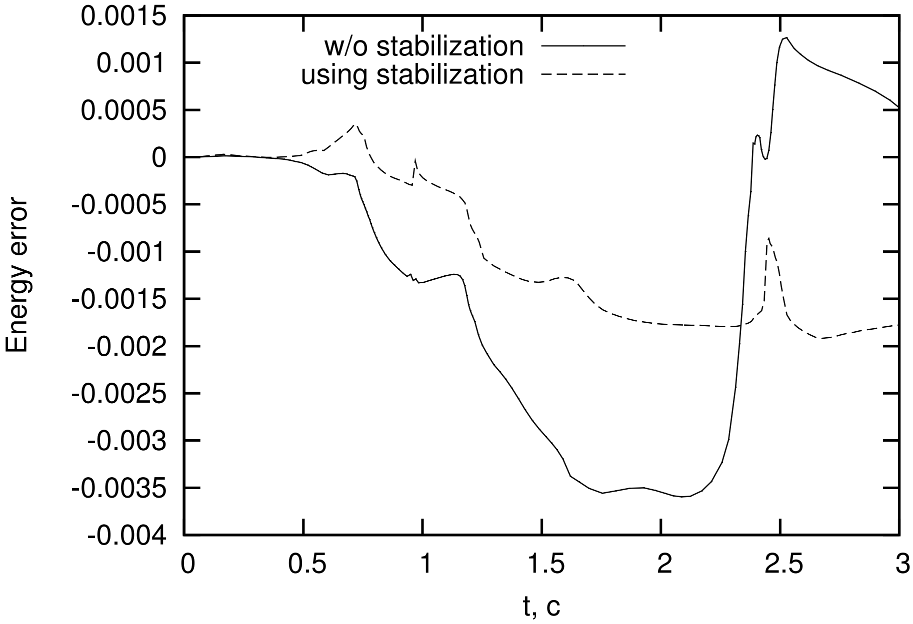

Figure 4. Energy error

booster thrust reduction leads to process in which booster slides over the core stage. Contact point slides several millimeters, than nozzle of gas-pressurized tank is opened. This nozzle create force that turns and completely separates first stage booster from core stage (fig. 6).

Let us write down the equations of this mechanical system. This model can be used for safety analysis of the system in depend on system parameters. For the sake of simplicity we write constraint equations for only one booster. At the first motion stage first stage booster can rotate around it's nose contact

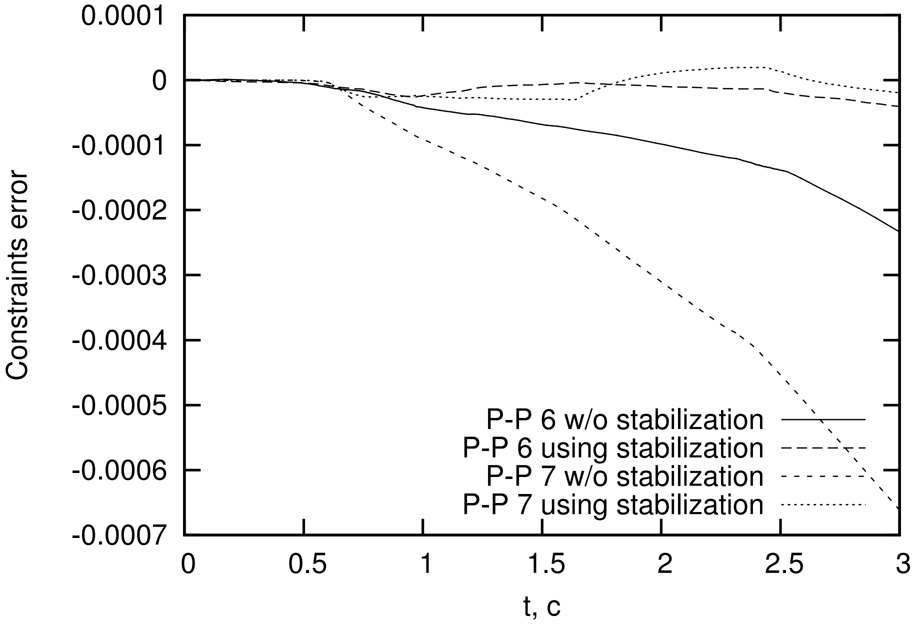

Figure 5. Constraints drift-off

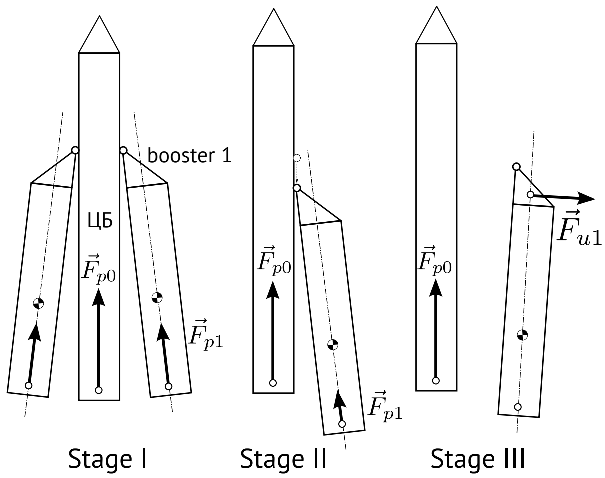

Figure 6. Motion phases of the first stage booster

point (fig. 6). We need three "point on plane" equations that are written as:

$$\begin{pmatrix} \mathbf{Q}_{c1} & \mathbf{Q}_{11} \\ \mathbf{Q}_{c2} & \mathbf{Q}_{12} \\ \mathbf{Q}_{c3} & \mathbf{Q}_{13} \end{pmatrix} \begin{pmatrix} \mathbf{X}_c \\ \mathbf{X}_1 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}. \tag{43}$$

$\ddot{\mathbf{X}}_c, \ddot{\mathbf{X}}_1$ - translational and rotational accelerations column matrices for booster and the core stage. Coefficient matrices (44) and (45) differs by unit vector $\mathbf{n}_1^{(c)}$, $\mathbf{n}_2^{(c)}$ and $\mathbf{n}_3^{(c)}$ only. These vectors determinate restricted directions for contact point motion at first stage (fig. 7) (the $\vec{n}_3$ vector is not shown at this picture).

$$\mathbf{Q}_{ck} = \left(-\mathbf{n}_k^{(c)T} \mathbf{A}^{cT} \middle| \mathbf{n}_k^{(c)T} \tilde{\rho}_c^{(c)} \right), \tag{44}$$

$$\mathbf{Q}_{1k} = \left(\mathbf{n}_k^{(c)T} \mathbf{A}^{cT} \middle| -\mathbf{n}_k^{(c)T} \mathbf{A}^{cT} \mathbf{A}^{1} \tilde{\rho}_1^{(1)} \right). \tag{45}$$

$$b_k = \mathbf{n}_k^{(c)T} (\tilde{\omega}_c^{(c)} \mathbf{A}^{cT} \dot{\rho}_{c1}^{(0)} - \tilde{\omega}_c^{(c)} \tilde{\omega}_c^{(c)} \mathbf{A}^{cT} \rho_{c1}^{(0)} \\ + \tilde{\omega}_c^{(c)} \mathbf{A}^{cT} \dot{\rho}_{c1}^{(0)} - \mathbf{A}^{cT} \mathbf{A}^{1} \tilde{\omega}_1^{(1)} \tilde{\omega}_1^{(1)} \rho_{1c}^{(1)}),\ k = 1, 2, 3. \tag{46}$$

$\omega_c^{(c)}$ - core stage angular velocity column matrix; $\omega_1^{(1)}$ - first stage booster angular velocity column matrix; $\rho_{c1}^{(c)}, \rho_{1c}^{(1)}$ - joint vectors in the core stage and in the booster frame respectively; $\mathbf{A}^c, \mathbf{A}^1$ - coordinate transformation matrices for the core stage and the booster. Reaction force acting on first stage booster

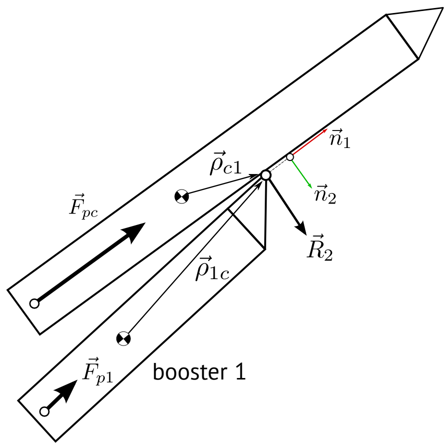

Figure 7. Contact between the first stage booster and the core stage

written as:

$$\mathbf{R}_k^{(0)} = -\mathbf{R}_{ck}^{(0)} = \mathbf{n}_k^{(c)T} \mathbf{A}^{1T} \lambda_k,\ k = 1, 2, 3.$$

here λ_k - is a Lagrange multiplier, that correspond to k constraint equation. Core stage experiences the reaction force acting in the opposite direction. Reaction force produces a torque $\mathbf{L}_{R_i}^{(1)}$:

$$\mathbf{L}_{R_k}^{(1)} = \mathbf{n}_k^{(c)T} \mathbf{A}^{cT} \mathbf{A}^{1} \tilde{\rho}_{1c}^{(1)} \lambda_i,\ i = 1, 2, 3.$$

Full equations set for first stage is written as:

$$\begin{pmatrix} \mathbf{M}_c & 0 & \mathbf{Q}_{c1}^T & \mathbf{Q}_{c2}^T & \mathbf{Q}_{c3}^T \\ 0 & \mathbf{M}_1 & \mathbf{Q}_{11}^T & \mathbf{Q}_{12}^T & \mathbf{Q}_{13}^T \\ \mathbf{Q}_{c1} & \mathbf{Q}_{11} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{Q}_{c2} & \mathbf{Q}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{Q}_{c3} & \mathbf{Q}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{X}_c \\ \mathbf{X}_1 \\ \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} = \begin{pmatrix} \mathbf{P}_c^{(0)} \\ \mathbf{P}_1^{(0)} \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} \tag{47}$$

where:

$$\mathbf{M}_c = \begin{pmatrix} \mathbf{I}_{3\times3}m_c & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_c^{(c)} \end{pmatrix},\ \mathbf{M}_1 = \begin{pmatrix} \mathbf{I}_{3\times3}m_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_1^{(1)} \end{pmatrix}$$

$$\mathbf{P}_c^{(0)} = \begin{pmatrix} \mathbf{A}^c\mathbf{F}_{pc}^{(c)} \\ \tilde{\omega}_c^{(c)}\mathbf{J}_c^{(c)}\omega_c^{(c)} \end{pmatrix},\ \mathbf{P}_1^{(0)} = \begin{pmatrix} \mathbf{A}^1\mathbf{F}_{p1}^{(1)} \\ \tilde{\omega}_1^{(1)}\mathbf{J}_1^{(1)}\omega_1^{(1)} \end{pmatrix}$$

Second stage of the motion starts when constraint force become less or equal than zero. During numerical integration algorithm should track the λ_1 value and stop integration process when:

$$\lambda_1 \leq 0.$$

For example in MATLAB environment it can be performed using 'event' option in the “odefile“. Second stage constraint equation system includes only two equations of (43) for $n_2^{(c)}$, $n_3^{(c)}$. Relative translation along vector $n_1^{(c)}$ is permissible: booster slides on the surface of the core stage. Third stage of the motion starts when constraint force become less or equal than zero, when booster detaches from the rocket:

$$\lambda_2 \leq 0.$$

Third stage equations are free of constraints equations.

At fig. 8 you can see temporal evolution of reaction forces R_1 and R_2. Reaction R_1 reaches zero at time about $0.7s$ when booster starts to slide. Later reaction R_2 reaches zero and after this event booster moves as a free body. On fig. 9 trajectorie of first stage boosters are showed.

6 Conclusion

Constraint equations form for mechanical systems with changing structure is proposed. Advantages of this equations form may be summarized as follows:

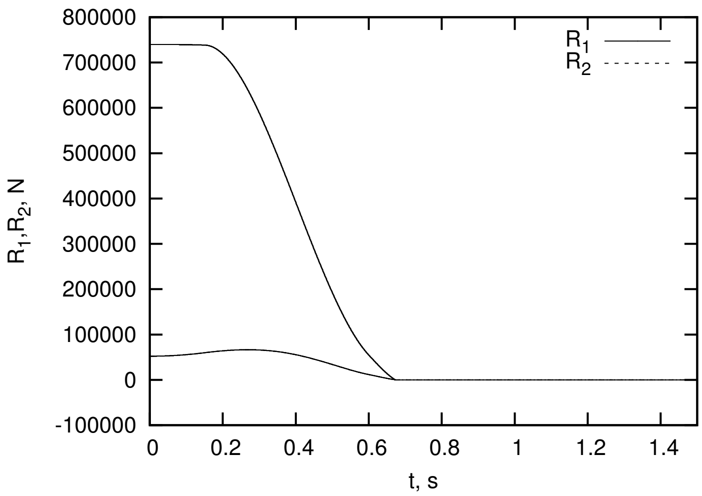

Figure 8. Reaction forces R_1 and R_2

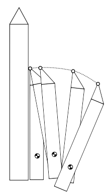

Figure 9. First stage booster trajectory

multibody structure change does not affect to variables set that describes system configuration, structure change is modeled by adding or removal simple constraint equations. Using proposed method models of rocket separation subsystem was build: separation process of first stage boosters that can be used for safety analysis of this mechanical system.

There are many commercial software for simulating multibody systems (e.g. MSC/ADAMS, “Universal mechanism“, “Euler”). In spite of this, proposed method offer several advantages over them. It provide full control over model building process that give assurance in results adequacy. Besides, proposed method can be easy implemented on free matrix-oriented software (OCTAVE, SciLab) and this method enables us to solve complex problems of mechanical systems with changing structure. Also it’s possible to use this method for educational purposes.

7 Acknowledgments

Support for this work was provided in part by Grant No. 09-01-0384 from the Russian Foundation for basic research.

References

[1] Y. Chen, V. Modi, C. De Silva, A. Misra, On the performance of variable geometry manipulators: part ii - three dimensional order n formulation, in: Acta Astronautica, Volume 49, Number 1, July 2001, Elsevier, New York, NY, USA, 2001, pp. 1–12. doi:http://doi.acm.org/10.1016/S0094-5765(01)00002-9.

[2] P. Williams, Dynamic multibody modeling for tethered space elevators, Acta Astronautica 65 (3-4) (2009) 399 – 422.

[3] H.-H. Zhang, H. Matsuo, Y. Morita, H. Yamakawa, Stabilization of a deployable flexible structure, Acta Astronautica 43 (7-8) (1998) 369 – 376.

[4] J. Unda, J. Weisz, J. Rivacoba, I. Urhen, R. Capitanio, Family of deployable/retractable structures for space application, Acta Astronautica 32 (12) (1994) 767 – 784.

[5] B. A. Boutin, A. K. Misra, V. J. Modi, Dynamics and control of variable-geometry truss structures, Acta Astronautica 45 (12) (1999) 717 – 728. doi:DOI: 10.1016/S0094-5765(99)00125-3.

[6] J. Y. S. Luh, M. W. Walker, R. P. C. Paul, On-line computational scheme for mechanical manipulators, J. DYN. SYS. MEAS. CONTR. 102 (2) (1980) 69–76.

[7] A. F. Vereshchagin, Computer simulation of the dynamics of complicated mechanisms of robotmanipulators, in: Engineering Cybernetics, no. 6, 1974, pp. 65–70.

[8] R. Featherstone, Robot dynamics: Equations and algorithms, in: IEEE Int. Conf. Robotics and Automation, 2000, pp. 826–834.

[9] R. Featherstone, Rigid Body Dynamics Algorithms, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[10] P. Santini, P. Gasbarri, General background and approach to multibody dynamics for space applications, 2009.

[11] P. Wijckmans, Conditioning of Differential Algebraic Equations and Numerical Solution of Multibody Dynamics, Prentice-Hall, 1996.

[12] J. Wittenburg, Dynamics of Systems of Rigid Bodies, B.G. Teubner Stuttgard, 1977.

[13] D. Baraff, Linear-time dynamics using lagrange multipliers, in: SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, ACM, New York, NY, USA, 1996, pp. 137–146.

[14] R. Featherstone, Efficient factorization of the joint-space inertia matrix for branched kinematic trees, Int. J. Rob. Res. 24 (6) (2005) 487–500. doi:http://dx.doi.org/10.1177/0278364905054928.

[15] A. A. Shabana, Computational dynamics, Wiley-IEEE, 2001.

[16] J. Baumgarte, Stabilization of constraints and integrals of motions in dynamical systems, Comp.Methods in Appl. Mech. and Eng. 1 (1972) 1–16.

[17] P. Lotstedt, On the relation between singular perturbation problems and differentialalgebraic equations, Report 100, Uppsala Univ., Dept. of Computer Sc. (1985).