How One Can Simulate Dynamics of Rolling Bodies via Dymola: Approach to Model Multibody System Dynamics Using Modelica

Ivan I. Kossenko and Maia S. Stavrovskaia
MSUS, Moscow Region, Russia, kosenko@ccas.ru

Abstract

An attempt to build more accurately the method to describe dynamics of multibody system (MBS) by means of Modelica is undertaken. In frame of the method under consideration can be simulated constraints of different types: holonomic/nonholonomic, scleronomic/rheonomic.

The model of a constraint allows to isolate mutually behavioral descriptions based on differential and algebraic equations correspondingly.

To illustrate an approach being applied the implementation of a constraint for bodies, rolling one relative to another is described. As an example the model of rattleback rolling on horizontal surface is investigated.

1 Preliminaries

A lot of methods to describe the structure of a multibody system using different graph approaches are known. See for instance [1, 2, 3], and further references one can find there. Usually MBS is assumed to consist of rigid bodies. Note that in the frame of a bondgraph approach the background of an energy interchanges is used [4].

When implementing the MBS structural analysis based on a force interactions either oriented or nonoriented graphs are used in dependence of the problems to be resolved. Newton's laws [5] allow to describe dynamics within the so called Newton's viewpoint. In such a way the translational–rotational motion of each body is described by the system of Newton–Euler's ODEs. The graph structure is derived from an analysis of mutual interactions for bodies the system composed from. Such an interactions is caused mainly by constraints. But there are cases of physical fields also occurred. In general, Newton's third law of dynamics implies a dual nature of interactions between the bodies.

Thus in a natural way from Newton's viewpoint the graph of an MBS structure is to be considered as a nonoriented one. In some particular cases the graph possesses special structure, and constraints are holonomic (i. e. integrable). Such situation occurs for instance in robotics where the structure is a tree. This fact used to reduce the source Newton–Euler system of ODEs with an attached subsystem of algebraic equations to some special kind of purely differential equations, for example of Lagrange ones. In this case natural approach assumes association of each dynamical ODE of the second order with the object comprising usually joint corresponding to a generalized coordinate, and an appropriate generalized force. Both usual linear force and torque of a couple can be in use. This force mainly is control one. It arises due to drive located at a joint. The solution of a kind previously described has been used in current Modelica MBS library of classes. For instance one can find such an approach in `Revolute` model where application of d'Alembert's principle relative to the revolution axis in behavioral section is equivalent to use of one second order ODE from Lagrange's equations for the whole holonomic mechanical system.

In general case the situation is more complicated, especially if nonholonomic (i. e. nonintegrable) constraints are used. Modeler has to take into account equations of constraints of algebraic, or even transcendental type. Fortunately today there exists background to build the models mentioned above, in particular: algorithms [6], modeling languages [7], and compilers [8]. To describe the models of an MBS we start from: (a) object–oriented paradigm [9] on one hand, and (b) so called physical principles of modeling [10] on the other one. Note that in our case of MBS dynamics one can consider in a natural way the rigid body as a main physical entity of the problems to be simulated.

1.1 Formal Description

Consider an MBS consisting of $m+1$ bodies $B_0, \ldots, B_m$. Represent it as a set $\mathcal{B} = \{B_0, \ldots, B_m\}$. Here B_0 is assumed to be a base body. We suppose B_0 to be connected with an inertial frame of reference, or to have a known motion with respect to the inertial frame of reference. For example one can imagine the base body as a rotating platform, or as a vehicle performing its motion according to a given law.

Some bodies are considered as connected by mechanical constraints. But in general it is not necessarily. Suppose all constraints compose the set $\mathcal{C} = \{C_1, \ldots, C_n\}$. We include in our considerations constraints of the following types: holonomic/nonholonomic, scleronomic/rheonomic. The latter properties mean the constraints having stationary/nonstationary parameters. For example one could consider the surface moving according to a prescribed law as a rheonomic constraint.

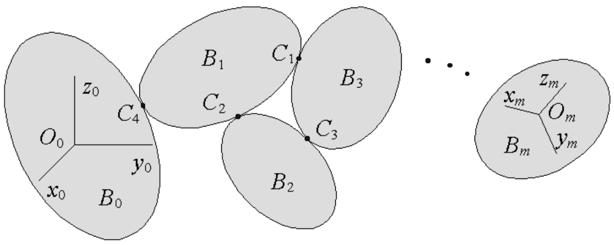

Figure 1.1: Multibody System

Thus one can uniquely represent a structure of the MBS via an nonoriented graph $G = (\mathcal{B}, \mathcal{C}, I)$. Here $I \subset \mathcal{C} \times \mathcal{B}$ is an incidence relation setting in a correspondence the vertex incident to every edge $C_i \in \mathcal{C}$ of the graph. According to physical reasons it is easy to see that for any mechanical constraint C_i there exist exactly two bodies $B_k, B_l \in \mathcal{B}$ connected by this constraint.

The incidence relation generates an adjacency relation $\mathcal{S} \subset \mathcal{B}^2$ on the set of vertices. In our case this relation has the properties: (a) antireflexiveness: a body isn't be connected with itself; (b) symmetry: because of the graph is nonoriented if $(B_k, B_l) \in \mathcal{S}$, then $(B_l, B_k) \in \mathcal{S}$.

1.2 Architecture of Bodies Mutual Interactions in MBS

It is clear that consideration of the graph G does not provide a structural information sufficient for the MBS dynamics description. Indeed, in addition to the force interaction represented usually by wrenches between bodies B_k, B_l through the constraint C_i there exist kinematical conditions specific for different kinds of constraints. In turn wrenches themselves can be represented by constraint forces and constraint torques couples. These forces and couples are connected by virtue of Newton's third law of dynamics.

Thus if the system of ODEs for translational–rotational motion can be associated with the object of a model corresponding to rigid body, then the system of the algebraic equations can be naturally associated with the object of a model corresponding to constraint. Note that according to consideration fulfilled above the set of algebraic equations comprises relations for constraint forces, torques of couples, and kinematical relations depending on kind of constraints. For such approach the differential and algebraic equations are said incapsulated in behavioral sections of the mentioned objects.

Thus all the "population" of any MBS model is reduced to objects of two classes: "Body" (objects $B_0, \ldots, B_m$), "Constraint" (objects $C_1, \ldots, C_n$). According to this approach simulating of the whole system behavior is reduced to permament informational interaction between the objects of two considered types. Within the frame of Newton's laws of dynamics one can construct the MBS as a communicative network for this interaction. In this case the objects of bodies "feel" the action of other ones through corresponding objects of constraints.

Physical interactions are conducted in models due to objects splitted also in two classes of ports: "Wrench Port", "Kinematical Port". The first one is to be used to transfer vectors of force, and torque of couple. In addition, "Wrench Port" has to be used for transferring the information about a current location of the point constraint force acts upon.

Remark 1.1 *In our idealized model the force interaction between bodies is realized at a geometric point. Its coordinates are fed outside constraint object through "Wrench Port" permanently in time.*

"Kinematical Port" is to be used to transfer the data of rigid body kinematics: configuration (position of center of mass, orientation), velocity (velocity of the center of mass, angular rate), and acceleration (acceleration of the center of mass, angular acceleration). Objects of classes "Body" and "Constraint" work as it is represented in Figure 1.2. The certain duality in a behavior of these objects can be easily observed.

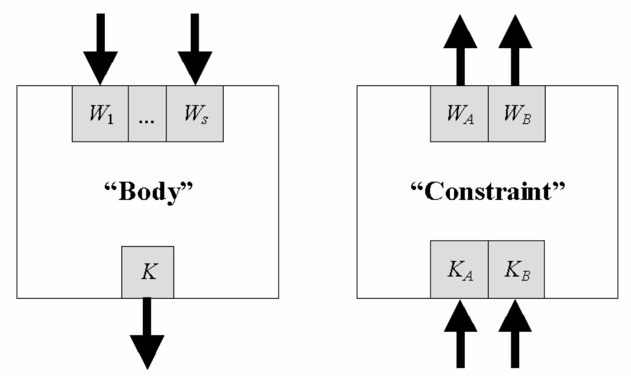

Figure 1.2: Typical Objects of MBS

Indeed, when getting force information through the ports $W_1, \ldots, W_s$ from the incident objects of a class "Constraint" the object of a class "Body" simultaneously generates, due to an integrator, kinematical information feeding outside through the port K. On the other hand every object of a class "Constraint" gets kinematical data from the objects corresponding to bodies connected by the constraint under consideration through its two "input" ports K_A, K_B. Simultaneously using the system of algebraic equations this object generates information concerning forces and torques of couples, and transmits the data to "output" ports W_A, W_B for the further transfer to objects of bodies under constraint.

According to classification of communicative ports performed above one can similarly classify the connectors used as "cables" for two purposes: (a) to transfer information about forces and torques; (b) to transfer the kinematical information.

Now it is possible to describe the architecture of information interactions within the particular constraint C_i corresponding to an individual edge of graph G, see Figure 1.3.

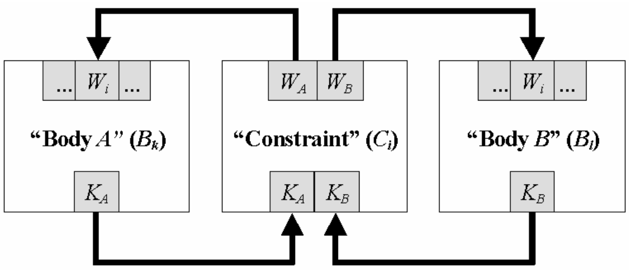

Figure 1.3: Architecture of Constraint

One should consider all connectors used above as bidirected ones. Arrows in Figure 1.3 are used to show the semantics of interactions. It's clear that the whole construction considered above is a virtual one. Constructing the model the compiler extracts all equations from the objects and assembles them composing the DAE system optimized for a numeric integrator.

As usual physical fields one can implement by applying of **`inner`** and **`outer`** specifications.

2 Rolling of Rigid Bodies

It turned out that the attempts to treat problems of nonholonomic mechanics within existing MultiBody classes library are not effective. Indeed, this library has been developed mainly for modeling of controlled motion in mechatronics and theory of machines and mechanisms. The case of rolling bodies, typical for nonholonomic mechanics can't be inserted in the formalism of joints and cuts in bodies. Here the position of a point of interaction between the bodies depends on dynamics of MBS.

Moreover, the situation turns being even more complicated if the friction of different kinds is taken into account, because relative sliding of bodies, unilateral motions, and impacts are assumed being allowed. To describe the dynamics of phenomena enumerated above one can apply well–known methods of classical mechanics staying simultaneously on positions of physical objects modeling. We mean differential equations of tranlational–rotational motion for interacting bodies, known as Newton–Euler's equations.

2.1 General Description

As an example for formal approach discussed above let us consider the problem on description of one body contiguous to another one. Such approach can be used by designer in order to avoid derivation of dynamic equations both for holonomic and nonholonomic mechanical systems. In the second case problem itself may be complicated enough. Note that traditional cuts, flanges, or joints as constraint interfaces seem to be impossible for use in the situation under consideration. In addition, one should take into account a useful property of mutual isolation of differential and algebraic equations incapsulated in the classes of types "Body" and "Constraint" correspondingly.

Consider a local fragment of a mechanical system, and suppose that this fragment consists of a pair of rigid bodies rolling one upon another. Then a general schema in Figure 1.3 is satisfied. Suppose that all force and kinematical data to be transferred through ports are represented in a unified way: relative to the base frame $O_0x_0y_0z_0$ of a reference connected with the body B_0. Wrench port consists of three arrays $\mathbf{P}, \mathbf{F}, \mathbf{M} \in \mathbf{R}^3$. Here $\mathbf{P}$ is an array of coordinates for the point of a contact between two bodies under constraint, $\mathbf{F}$ is a constraint force vector, $\mathbf{M}$ is a constraint couple torque vector. In general case components of the array $\mathbf{P}$ are computed in the object of a constraint. The vectors $\mathbf{F}$, $\mathbf{M}$ are assumed expressing the "action" of constraint object to body object, of course in a virtual sense. The class to transfer force information can read:

```
connector WrenchPort
  SI.Position P[3];
  SI.Force F[3];
  SI.Torque M[3];
end WrenchPort;
```

The kinematical port consists of six arrays: $\mathbf{r}, \mathbf{v}, \mathbf{a}, \omega, \varepsilon \in \mathbf{R}^3$, $T \in SO(3)$. The array $\mathbf{r}$ corresponds to the radius–vector of the mass center of the body, $\mathbf{v}$ corresponds to the velocity of this point, $\mathbf{a}$ corresponds to its acceleration. T is an orthogonal matrix of a current body orientation. The columns of the matrix T consist of projections of unit vectors of the orthonormal base connected with a moving body into the axes of the base body frame. The class "Kinematical Port" in Modelica can be defined as:

```
connector KinematicPort
  SI.Position r[3];
  SI.Velocity v[3];
  SI.Acceleration a[3];
  SI.Real T[3,3];
  SI.AngularVelocity omega[3];
  SI.AngularAcceleration epsilon[3];
end KinematicPort;
```

All the objects of a class "Constraint" must have classes–inheritors as subtypes of a corresponding superclass. According to Newton's third law this superclass must contain the equations of the form

$$\mathbf{F}_A + \mathbf{F}_B = \mathbf{0}, \quad \mathbf{M}_A + \mathbf{M}_B = \mathbf{0}. \tag{2.1}$$

in its behavioral section. Here arrays $\mathbf{F}_A$, $\mathbf{M}_A$ and $\mathbf{F}_B$, $\mathbf{M}_B$ represent constraint forces and torques "acting in directions" of bodies A and B correspondingly. Kinematical equations for different types of constraints are to be added to equations (2.1) in different classes–inheritors corresponding to these particular types of constraints.

Properties (2.1) usually conducted through **flow**–variables are implemented here in a natural way in the behavioral section of the base superclass for mechanical constraints. They read:

```
partial model Constraint
  KinematicPort InPortA;
  WrenchPort OutPortA;
  KinematicPort InPortB;
  WrenchPort OutPortB;
equation
  OutPortA.F + OutPortB.F = {0,0,0};
  OutPortA.M + OutPortB.M = {0,0,0};
end Constraint;
```

Remark 2.1 *Model developer can create classes of complicated types of constraints such that equations (2.1) are not satisfied. For example such a constraint one can imagine as a thread thrown over the pulley, see Figure 2.1. It is clear that this constraint can be decomposed to components in such a way that the equations (2.1) are satisfied for each elementary constraint. However in applications it is often suitable to deal with constraints of a complex, combined type directly.*

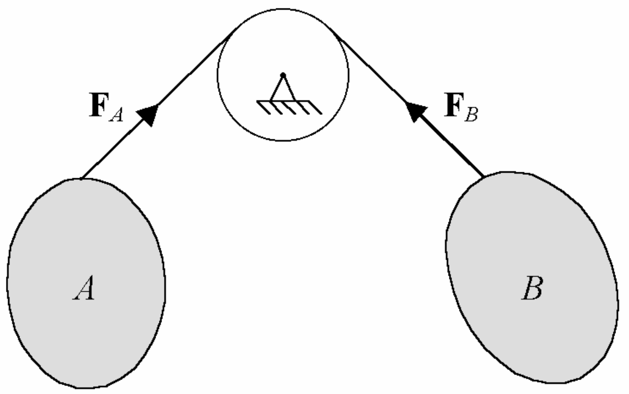

Figure 2.1: Example to Remark 2.1

Now start to construct behavioral equations for the object simulating of a constraint of the rolling type, see Figure 2.2. First of all let us describe the system of equations defining the position of a contact point. A constraint object has to "know", i. e. to incapsulate inside itself the equations for contiguous surfaces

$$f_A(x_k, y_k, z_k) = 0, \quad f_B(x_l, y_l, z_l) = 0.$$

Here all equations are defined with respect to the frame fixed in a corresponding body. Suppose that for the instant t these surfaces are described by equations

$$g_A(x_0, y_0, z_0) = 0, \quad g_B(x_0, y_0, z_0) = 0$$

written in the base frame. Here the expressions for the functions g_A and g_B can be easily obtained as

$$\begin{array}{rcl} g_A(\mathbf{r}_0) & = & f_A\left[T_k^{-1}(\mathbf{r}_0 - \mathbf{r}_{O_k})\right], \\ g_B(\mathbf{r}_0) & = & f_B\left[T_l^{-1}(\mathbf{r}_0 - \mathbf{r}_{O_l})\right]. \end{array}$$

Here for sake of brevity we denote $\mathbf{r}_0 = (x_0, y_0, z_0)^T$. The vectors $\mathbf{r}_{O_k}$, $\mathbf{r}_{O_l}$ determine mass center positions for the connected bodies. All radius vectors $\mathbf{r}_0$, $\mathbf{r}_{O_k}$, $\mathbf{r}_{O_l}$ are assumed being represented in the base frame. Note that computation of matrices inverse to orthogonal ones is reduced simply to matrix transposition.

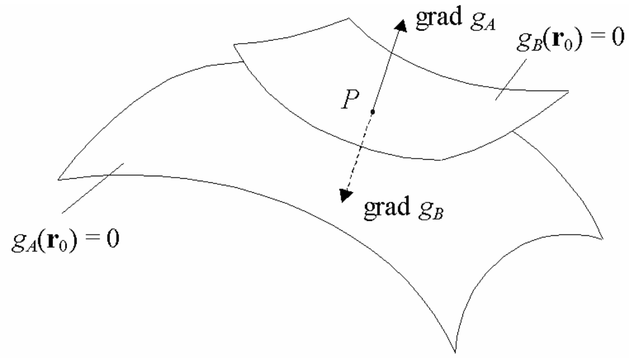

Figure 2.2: Vicinity of Contact Point

When rolling, the surfaces touch each other at the point P which is to be found, see Figure 2.2. The necessary condition of tangency reads

$$\mathrm{grad}\, g_A = \lambda \cdot \mathrm{grad}\, g_B. \tag{2.2}$$

Here λ is a scalar factor playing a role of additional auxiliary variable. In general position the system (2.2) defines uniquely a curve consisting of points in which the surfaces

$$g_A(\mathbf{r}_0) = \alpha, \quad g_B(\mathbf{r}_0) = \beta$$

have common tangent planes. One can consider λ as a coordinate on the curve. In general position such a curve intersects the surface

$$g_A(\mathbf{r}_0) = 0, \quad \text{or} \quad g_B(\mathbf{r}_0) = 0 \tag{2.3}$$

transversally. Thus the system of equations to find a contact point can be composed using (2.2) and (2.3). Here one has four scalar equations and four unknown variables: x_0, y_0, z_0, λ.

To complete the process of constructing the class "Roll", an inheritor of a superclass "Constraint" one should append the condition of absence of sliding at the point of a contact

$$\mathbf{v}_k + [\omega_k, \mathbf{r}_0 - \mathbf{r}_{O_k}] = \mathbf{v}_l + [\omega_l, \mathbf{r}_0 - \mathbf{r}_{O_l}] \tag{2.4}$$

to a system of equations (2.2), (2.3). Here vectors $\mathbf{v}_k$, ω_k, $\mathbf{v}_l$, ω_l denote linear velocities of mass centers, and angular velocities for the bodies subjected to constraint. Both sides of (2.4) are obtained from the rigid body kinematics [5]. Corresponding inheritor has the following Modelica code:

```
partial model Roll
  extends Constraint;
  SI.Position r[3];
equation
  InPortA.v +
  cross(InPortA.omega,r - InPortA.r) =
  InPortB.v +
  cross(InPortB.omega,r - InPortB.r);
  OutPortA.P = r;
  OutPortB.P = r;
  OutPortB.M = {0,0,0};
end Roll;
```

2.2 Dynamics of Rattleback

Further consider the simplified model of a rigid bodies rolling, namely dynamics of the rattleback on an immovable horizontal surface [11]. In this case the base body is supposed being fixed. Its boudary is a fixed horizontal plane which is considered as a surface for rolling. This body plays a role of the "Body A", see Figure 2.3. Thus it is the same body as above denoted by B_0. To describe its dynamics one need not the differential equations. All kinematical variables are zero–valued vectors. Matrix of orientation T_0 is an identity one.

Note objects of base bodies play a special role to describe the motion according to a predefined law. Corresponding classes have no any differential equations in their behavior. One can write down superclass of base body in the form:

```
partial model BaseBody
```

```
  KinematicPort OutPort;
end BaseBody;
```

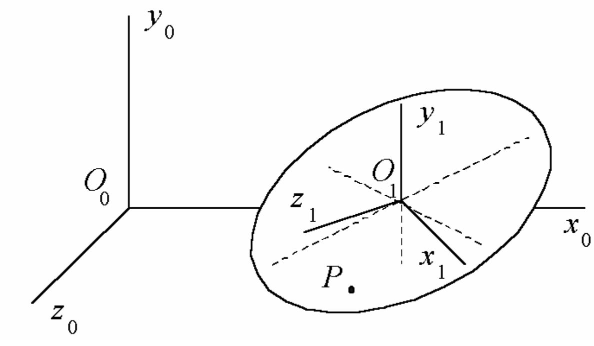

Figure 2.3: Rattleback on Horizontal Surface

Since body B_0 is supposed being fixed in the inertial frame then the class–inheritor can be described as:

```
model Base
  extends BaseBody;
  VisualShape Plane(
    r0={0,0,0},
    Shape="box",
    LengthDirection={0,-1,0},
    WidthDirection={1,0,0},
    Length=0.1,
    Width=10,
    Height=10,
    Material={0,0,1,0});
  WrenchPort InPortRoll;
equation
  OutPort.r = {0,0,0};
  OutPort.v = {0,0,0};
  OutPort.a = {0,0,0};
  OutPort.T = [1, 0, 0;
               0, 1, 0;
               0, 0, 1];
  OutPort.omega = {0,0,0};
  OutPort.epsilon = {0,0,0};
end Base;
```

The rattleback B_1 plays a role of the "Body B". Number of moving bodies in the MBS is $m = 1$. In superclass "Body", named in our package as `RigidBody` dynamics of rigid body is described here by means of Newton's differential equations for the body mass center, and by Euler's differential equations for rotational motion. The Euler equations are constructed using quaternion algebra [12] in a following way

$$\frac{d\mathbf{q}}{dt} = \frac{1}{2}\mathbf{q} \circ \begin{bmatrix} 0 \\ \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix}, \quad I\frac{d\Omega}{dt} + [\Omega, I\Omega] = \mathbf{N}.$$

Here first equation is kinematical one, and the second equation is one for dynamics of rigid body. Quaternion $\mathbf{q} = (q_1, q_2, q_3, q_4)^T \in \mathbf{H} \simeq \mathbf{R}^4$ uniquely defines rotational matrix R; the quaternion algebra $\mathbf{H}$ is considered as a linear space $\mathbf{R}^4$; the binary operation $\circ$ denotes quaternion multiplication. The matrix of inertia tensor I, the vector of angular velocity $\Omega = (\Omega_1, \Omega_2, \Omega_3)^T \in \mathbf{R}^3$, and vector of total torque $\mathbf{N}$ are considered with respect to central principal axes of inertia of the body. Hence before exporting of kinematical data from the object of class `RigidBody` one must perform the conversion $\omega = R\Omega$. Similarly total torque after importing and before using inside behavioral section also is to be tranformed $\mathbf{M} \mapsto \mathbf{N}$ according to the formula $\mathbf{N} = R^T\mathbf{M}$.

Usually the rattleback, or wobblestone, or Celtic stone is assumed being rigid body bounded by paraboloidal or ellipsoidal surface. This body is assumed possessing a central principal axes of inertia which are not collinear to body's axes of symmetry. Consider the case of an ellipsoidal surface.

Suppose that the central principal moments of inertia for the moving body read

$$I_{x_1x_1} = 2, \quad I_{y_1y_1} = 3, \quad I_{z_1z_1} = 1.$$

Fix also its mass value $\mu = 1$. To be definite one can suppose all physical measures based for instance on SI units. Then the rattleback model can be represented in the form:

```
model RollingBody
  extends RigidBody;
  outer Real[3] Gravity;
  // Ellipsoid semi-diameters
  outer SI.Length a1;
  outer SI.Length b1;
  outer SI.Length c1;
  VisualShape Body(
    r0=0,0,-c1,
    Shape="sphere",
    LengthDirection=0,0,1,
    WidthDirection=1,0,0,
    Length=2*c1,
    Width=2*a1,
    Height=2*b1,
    Material=1,0,0,1);
  SI.Energy E; // Full energy
  SI.Energy K1;// Kinetic energy
               // of translations
  SI.Energy K2;// Kinetic energy
               // of rotations
  SI.Energy P; // Potential energy
  WrenchPort InPortRoll;
```

```
equation
  F = m*Gravity + InPortRoll.F;
  M = InPortRoll.M +
      cross(InPortRoll.P - r,
            InPortRoll.F);
  Body.S = T;
  Body.r = r;
  K1 = 0.5*m*v*v;
  K2 = 0.5*omega*I*omega;
  P = -m*r*Gravity;
  E = K1 + K2 + P;
end RollingBody;
```

Now consider the building of a constraint. Since $f_A(x_0, y_0, z_0) \equiv y_0$ then grad $f_A = (0, 1, 0)^T$. A bounding surface for the body B is assumed to be of the ellipsoidal shape having the following semi–diameters $a_1 = 2$, $b_1 = 1$, $c_1 = 3$. The matrix of the corresponding quadratic form relative to principal axes of the ellipsoid reads

$$B_1 = \begin{pmatrix} a_1^{-2} & 0 & 0 \\ 0 & a_2^{-2} & 0 \\ 0 & 0 & a_3^{-2} \end{pmatrix}.$$

Suppose that the second principal axis directed along the axis O_1y_1 of the ellipsoid of inertia is coincident to the axis of the body surface. outer shape directed identically. Further, let us turn the second ellipsoid relative to the first one about O_1y_1 by an angle $\delta = \pi/10$. Then the matrix of a quadratic form for rolling ellipsoid relative to central principal–axis system has the form

$$B = RB_1R^T, \quad R = \begin{pmatrix} \cos\delta & 0 & \sin\delta \\ 0 & 1 & 0 \\ -\sin\delta & 0 & \cos\delta \end{pmatrix}.$$

The equations defining the position $\mathbf{r}_0 = \mathbf{r}_P$ of a contact point P read

$$\begin{aligned} (\text{grad}\, f_A, \mathbf{r}_P) &= 0, \\ \text{grad}\, f_A &= \lambda \cdot \left(T_1 \cdot B \cdot T_1^{-1}\right)(\mathbf{r}_0 - \mathbf{r}_{O_1}). \end{aligned} \tag{2.5}$$

According to previous considerations the system (2.5) comprises four scalar equations and four unknown values: x_P, y_P, z_P, λ. First equation can be easily reduced to the simple form $y_P = 0$. Finally, class–inheritor for the constraint under consideration takes the form:

```
model Ellipsoid_on_Plane
  extends Roll;
  outer SI.Length a1;
  outer SI.Length b1;
  outer SI.Length c1;
  outer SI.Angle delta;
  parameter Real R[3,3]=
    [cos(delta), 0, sin(delta);
     0, 1, 0;
     -sin(delta), 0, cos(delta)];
  parameter Real B1[3, 3]=
    [1/a1^2, 0, 0;
     0, 1/b1^2, 0;
     0, 0, 1/c1^2];
  parameter Real B[3,3]=
            R*B1*transpose(R);
  parameter Real n[3]=0,1,0;
  parameter SI.Length d=0;
  Real lambda;
equation
  n*r = d;
  n = lambda*
      InPortB.T*B*
      transpose(InPortB.T)*
      (r - InPortB.r);
end Ellipsoid_on_Plane;
```

Now we can compose the testbench model for simulation of dynamics of the rattleback as:

```
model Test
  parameter SI.Acceleration g=9.81;
  inner parameter SI.Acceleration[3]
                 Gravity={0,-g,0};
  inner parameter SI.Length a1=2;
  inner parameter SI.Length b1=1;
  inner parameter SI.Length c1=3;
  inner parameter SI.Angle delta=
          Modelica.Constants.pi/10;
  Base Base1;
  RollingBody RollingBody1(
    q(start={1,0,0,0}),// Initial
                       // quaternion
    r(start={0,1,0}),
    I=[2, 0, 0; 0, 3, 0; 0, 0, 1],
    v(start={0.05,0,0}),
    omega(start={0,-1,-0.05}));
  Ellipsoid_on_Plane
            Ellipsoid_on_Plane1;
equation
  connect(Base1.InPortRoll,
    Ellipsoid_on_Plane1.OutPortA);
  connect(Base1.OutPort,
    Ellipsoid_on_Plane1.InPortA);
  connect(
    Ellipsoid_on_Plane1.InPortB,
    RollingBody1.OutPort);
  connect(
    Ellipsoid_on_Plane1.OutPortB,
    RollingBody1.InPortRoll);
end Test;
```

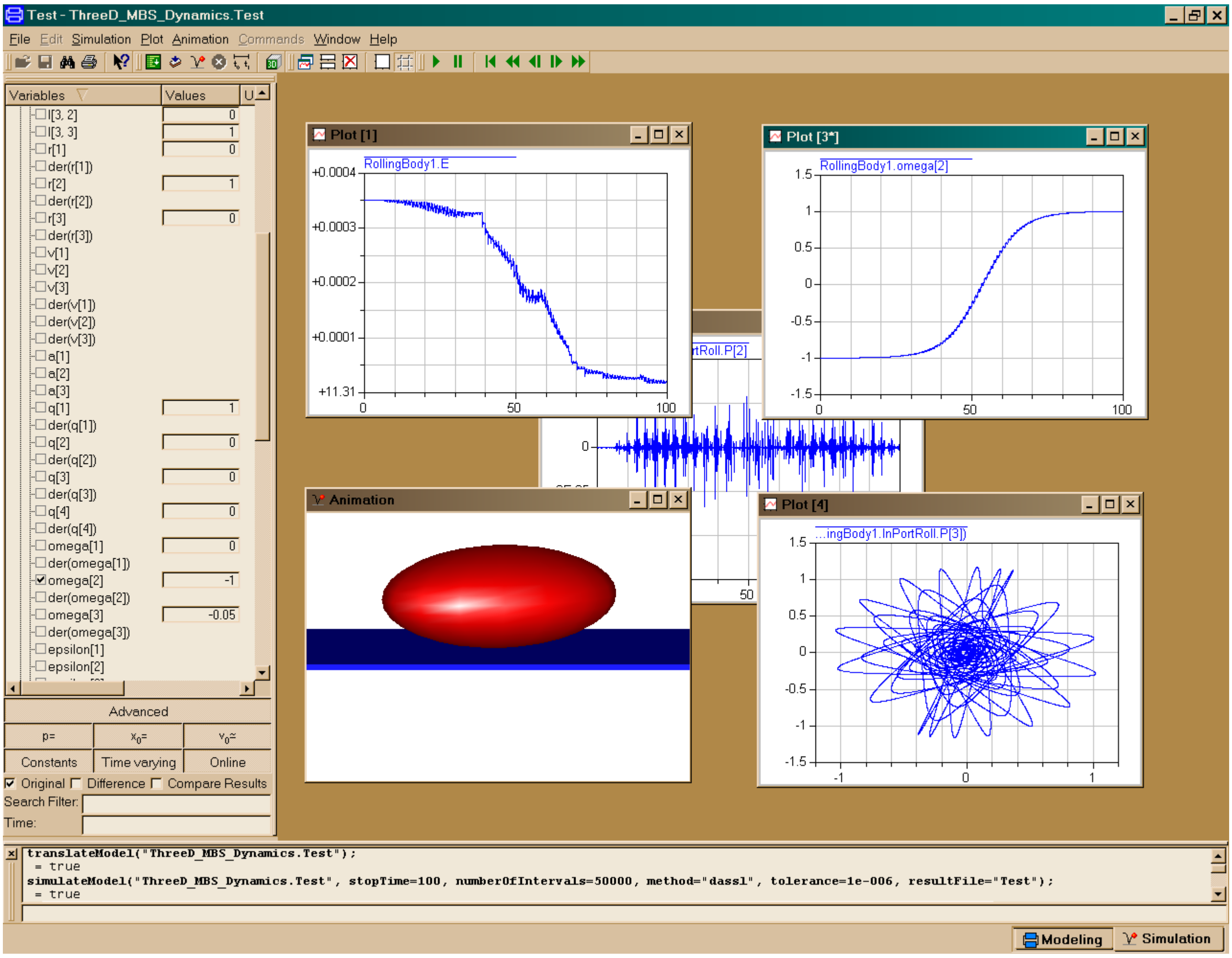

Figure 2.4: General View of Simulation Results

The model described above has been developed using Modelica language as a package. The high quality of an approximation for the rattleback motions has been verified through different simulations performed. For one of the model runs general view of simulation results is shown in Figure 2.4. Initial conditions are defined in a following way

$$\mathbf{r}_{O_1}(0) = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \qquad \mathbf{v}_1(0) = \begin{pmatrix} 0.05 \\ 0 \\ 0 \end{pmatrix},$$

$$T_1(0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \omega_1(0) = \begin{pmatrix} 0 \\ -1 \\ -0.05 \end{pmatrix}.$$

Integral of energy value was under monitoring see Figure 2.5. One can observe for this value an extremely slow drift: height of the whole plot equals to 0.0004 while base value is equal to 11.31 units of energy. Obviously such a drift is caused by computational errors.

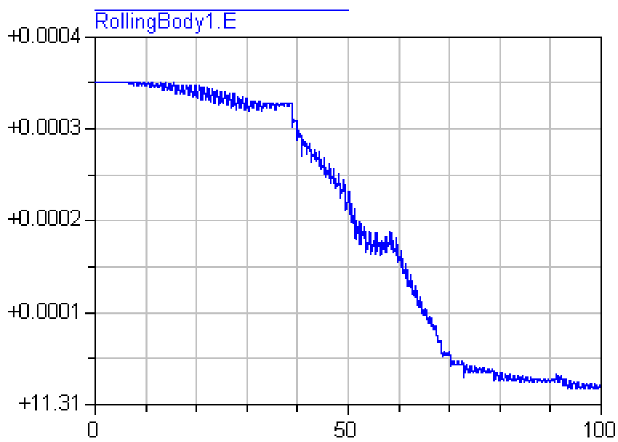

Figure 2.5: Preservation of Energy

Trajectory of a contact point in plane z_0x_0, see Figure 2.6 was also under monitoring. The constraint is satisfied with high accuracy permanently for all instants of simulation time. Indeed, such an accuracy can be investigated using variable y_P from the equa-

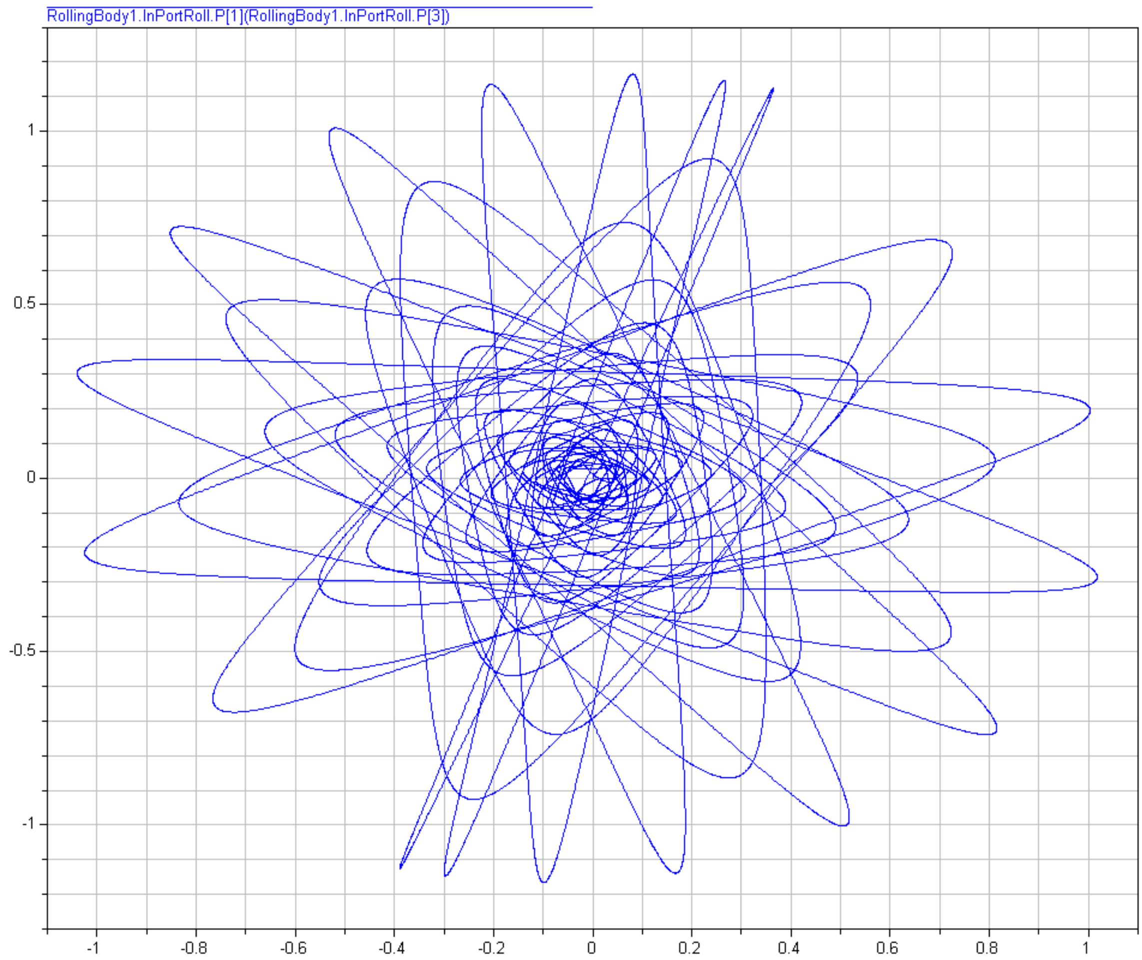

Figure 2.6: Contact Point Trajectory

tions (2.5). In Figure 2.7 we see that the function $y_P(t)$ performs only noisy oscillations almost vanishing near its zero value. Here height of the plot equals to 10^{-24} of length unit.

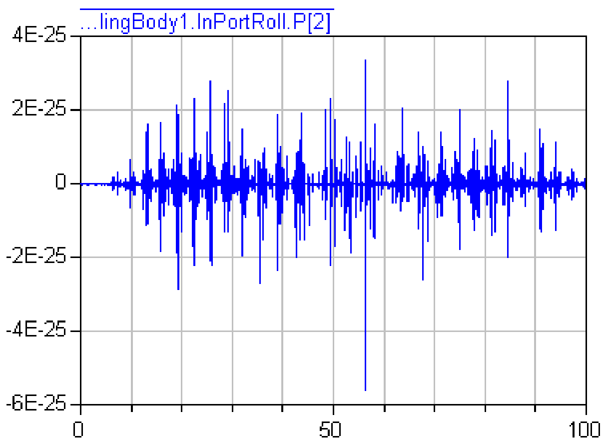

Figure 2.7: Preservation of Constraint Accuracy

Due to high quality of a numeric model one can easily observe the known dynamical properties of the rattleback. These latter cause in particular change of direction of angular velocity vector corresponding to rotation about central principal axis O_1y_1 of Celtic stone, see Figure 2.8. Initially axis O_1y_1 is directed vertically downwards, and rattleback rotates clockwise. Initial direction of the angular velocity slightly deviates from the local vertical. Then when time passes value of $t = 50$ units vertical component of angular velocity passes through its zero value, and one can observe wobbling motions of the rattleback. One instant of such wobbling is fixed in Figure 2.9. Note that the total energy is a constant because the mechanical system under consideration is conservative one. Then rotation becomes almost permanent but now counterclockwise. It easy to see (in Figure 2.8) that the angular velocity projection onto inertial axis y_0 is scaled from -1 to 1 during time of simulation. But its value undergoes slight oscillations of several frequencies.

Using visual environment of Dymola one can also easy build 3D–animation of the rattleback rolling on

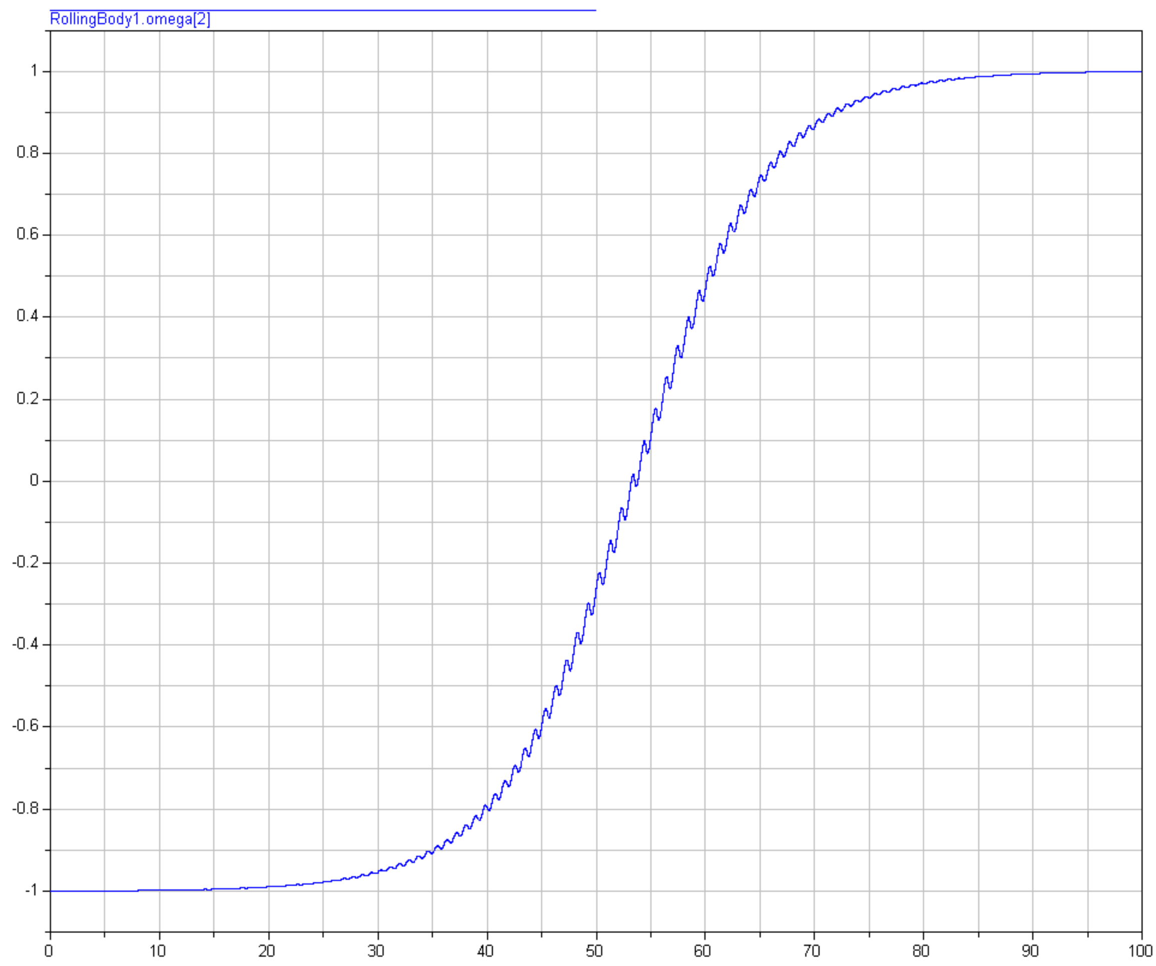

Figure 2.8: Behavior of Vertical Component of Angular Rate

a plane, see for example Figure 2.9.

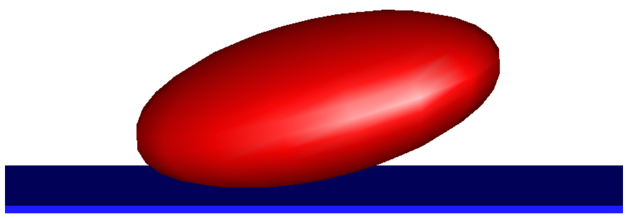

Figure 2.9: Instant Shot at a Moment When Stone Is Prepared to Change Orientation of Its Rotation

3 Directions of Further Development

Development of the Modelica code similar to one presented above opens a wide range of possibilities to model easily complicated problems of MBS dynamics. Among them: (a) dynamics of systems with sliding subjected to friction of various kinds; (b) dynamics of systems subjected to unilateral constraints with impacts [13]. In both cases to realize models one can apply Modelica's **when** clause in behavioral section. Then different cases of sliding and friction correspond to different cases of equations for forces and torques. For instance in simplest cases numerical models of tops rolling/sliding on surfaces can be investigated as exercises. Note that modeling of dynamics with unilateral constraints is also convenient for Modelica use because of its facilities for events processing. In all cases we deal with dynamics arranged as a piecewise smooth motion.

It should be also interesting to construct realistic model of dynamics for the truck rolling on a road of different surface quality in different weather conditions, and a model of dynamics for a heap consisting of contacting bodies, etc.

Returning to the problem considered above note that the model also simplifies qualitative dynamical analysis for long time simulations. An existence of such structures in phase space like attractors in dynamics of the rattleback [14] can be demonstrated.

4 Conclusion

Computations corresponding to case of Kane and Levinson have been performed. Results of simulations are identical in all details. Moreover, no special differential equations for dynamics of nonholonomic systems have been used! This is a real way to achieve a unified approach to modeling of both holonomic and nonholonomic MBS. Compiler itself incapsulates implicitly the use of equations of motion for nonholonomic mechanical system in its algorithm. Thus the known problem of ODEs derivation for such systems sometimes nontrivial and difficult seems to be overcome in automatic mode, at least for problems with algebraic (nontrancedental) constraints, and for cases of so called general position.

5 Acknowledgement

The paper was prepared with partial support of Russian Foundation for Basic Research, grants 02–01–00196, SS–2000.2003.1.

References

[1] Wittenburg, J., Dynamics of Systems of Rigid Bodies. — Stuttgart: B. G. Teubner, 1977.

[2] von Schwerin, R., Multi Body System SIMulation. Numerical Methods, Algorithms, and Software. Lecture Notes in Computational Science and Engineering, Vol. 7. — Berlin Heidelberg: Springer, 1999.

[3] Sinha, R., Paredis, C. J. J., Liang, V-C., and Khosla, P. K., Modeling and Simulation Methods for Design of Engineering Systems. // Journal of Computing and Information Science in Engineering, 2001, Vol. 1, Iss. 1, pp. 84–91.

[4] Mukherjee, A., Karmakar, R., Modelling and Simulation of Engineering Systems through Bondgraphs. — Alpha Science International Ltd., 2000.

[5] Routh, E. J., A Treatise on the Dynamics of a System of Rigid Bodies. — London: Vol. 1, 1897; Vol. 2, 1905.

[6] Ascher, U. M., Petzold, L. R., Computer Methods for Ordinary Differential Equations and Differential–Algebraic Equations. — SIAM, 1998.

[7] Modelica — A Unified Object-Oriented Language for Physical Systems Modeling. Tutorial. — Modelica Association, 2000.

[8] Dymola. Dynamic Modeling Laboratory. User's Manual. Version 5.0a — Lund: Dynasim AB, Research Park Ideon, 2002.

[9] Booch, G., Object–Oriented Analysis and Design with Applications. — Addison–Wesley Longman Inc. 1994.

[10] Cellier, F. E., Elmqvist, H., Otter, M., Modeling from Physical Principles. // in: Levine, W. S. (Ed.), The Control Handbook. — Boca Raton, FL: CRC Press, 1996. — pp. 99–107.

[11] Kane, T. R., Levinson, D. A., Realistic Mathematical Modeling of the Rattleback. // International Journal of Non–Linear Mechanics, 1982, Vol. 17, Iss. 3, pp. 175–186.

[12] Kosenko, I. I., Integration of the Equations of the Rotational Motion of a Rigid Body in the Quaternion Algebra. The Euler Case. // Journal of Applied Mathematics and Mechanics, 1998, Vol. 62, Iss. 2, pp. 193–200.

[13] Pfeiffer, F., Unilateral Multibody Dynamics. // Meccanica, 1999, Vol. 34, No. 6, pp. 437–451.

[14] Borisov, A. V., and Mamaev, I. S., Strange Attractors in Rattleback Dynamics // Physics–Uspekhi, 2003, Vol. 46, No. 4, pp. 393–403.