UNIVERSAL MECHANISM 9

universal mechanism

User`s manual

Collection of Models

2020

Contents

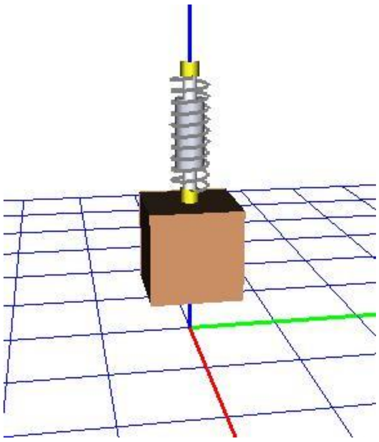
7. Collection of models

7.1. General information

The manual is intended for help in learning different elements (joints, force elements) as well as some features of creating UM models, parameterization of models as well as analysis of models with UM. The manual is based on ready models, which are located in the {UM Data}\SAMPLES\LIBRARY directory.

7.2. Damped free vibrations

Example: illustration to the 'damping ratio of critical' β (Chapter 2. Sect. *Methodology of choice of contact parameters*).



UM model: LIBRARY\DampingRatio.

7.2.1. Creating model

In this section we consider a very simple model – a body moving vertically under the action of gravity and linear viscoelastic force acting in the vertical direction. Let us consider modeling force elements.

Mechanical system includes the only force – bipolar force of **Expression** type. **Expression** is given as **-cstiff*(x-1.25)-cdiss*v-mass*9.81**.

Let us consider the expression in detail. It consists of three summands. The first one corresponds to an elastic force, the second one is dissipation, and the third on is a constant equal to the body weight. The last term ensures zero value of the force when the coordinates and velocities are zeroes.

x,v are the standard identifiers for the length of elements and its time derivative. The identifiers *cstiff*, *cdiss* parameterize stiffness and damping coefficients. The number 1.25 is equal to the length of the undeformed element and corresponds to the value of the *Length* parameter computed by the program. So the body is at the equilibrium position.

On this step of the model description is quite not important what numeric values initialize the identifiers *cstiff, cdiss*), because they are expressed via other identifiers: frequency f (in Hz) and damping ratio beta: **cstiff = 4*pi^2*f^2*Mass, cdiss = 2*beta*sqrt(mass*cstiff)**.

Here we used formulas obtained in Chapter 2, Sect. *Methodology of choice of contact parameters*.

If we set now some nonzero values for the frequency and damping ratio (*f*=1, *beta*=0.1), the value of the stiffness and damping coefficients are computed automatically.

Name	Expression	Value	Com
mass	1000		
f	1		
cstiff	4*pi^2*f^2*Mass	3.94784E+4	
beta	0.1		
cdiss	2*beta*sqrt(mas	1256.64	

Figure 7.1. Automatic computing the *cstiff, cdiss* identifiers

Note. The body has six degrees of freedom because in the case of visual adding bodies, joints with 6 d.o.f. are assigned to each of them automatically (3 Cartesian coordinates and 3 angles of rotation in the sequence 1,2,3 – Cardan angles). This joint is not added to the list of joints, and the user does not watch it, but the corresponding degrees of freedom are available for changing at the *Position* tab of the body description window as well as at simulation. If an additional joint describing position of the body relatively to the base or to one of other bodies is introduced, the described six degrees of freedom are ignored.

7.2.2. Simulation

The model above coincides fully with the model LIBRARY\DampingRatio, so the further results are obtained with the model from the library. Open the model in the Simulation module. A standard configuration of the model contains one animation and one graphic window. The graphic window contains one variable – Z-coordinate of the box center of gravity. In the initial state the box is located 0.5m over the equilibrium position. Make several numeric experiments alternating damping and frequency (identifiers *beta, f*).

Note, that the model is used for obtaining the plot of damped vibrations, Chapter 2, Sect. *Methodology of choice of contact parameters, figure Damped vibrations for various damping ratios*.

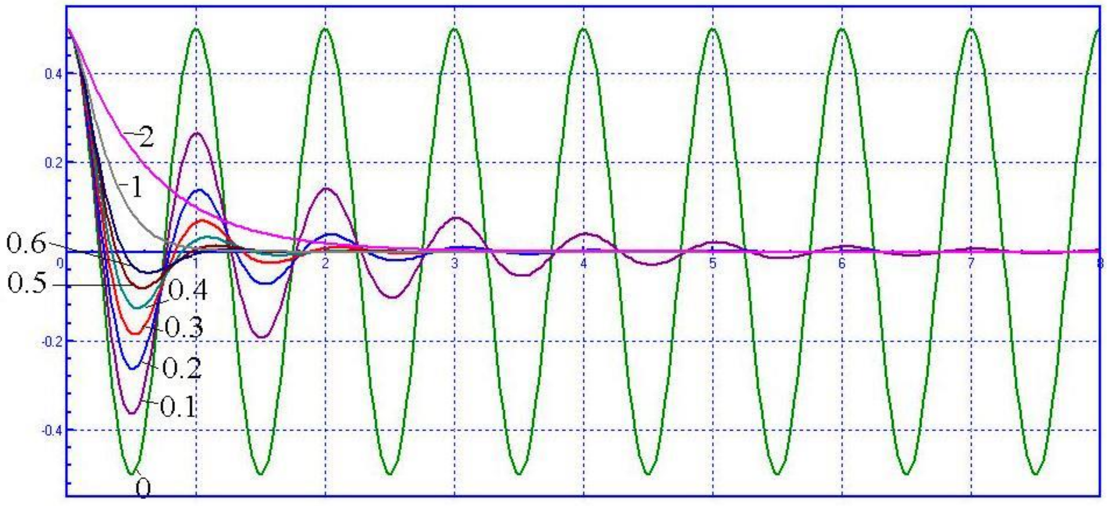
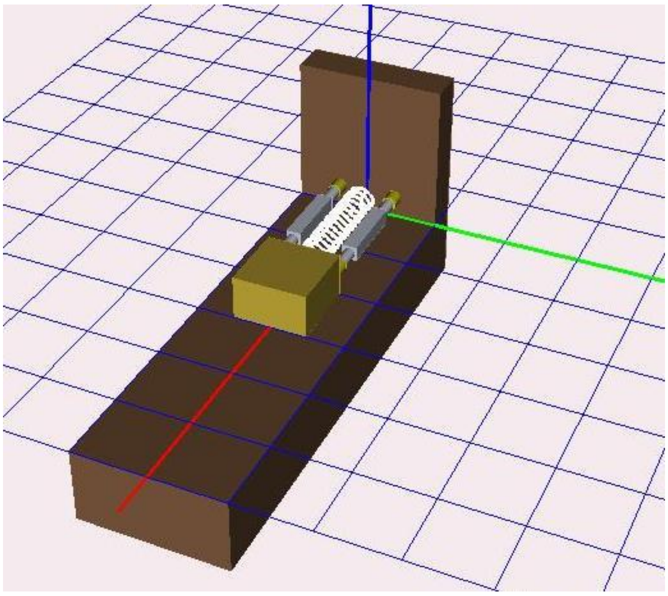


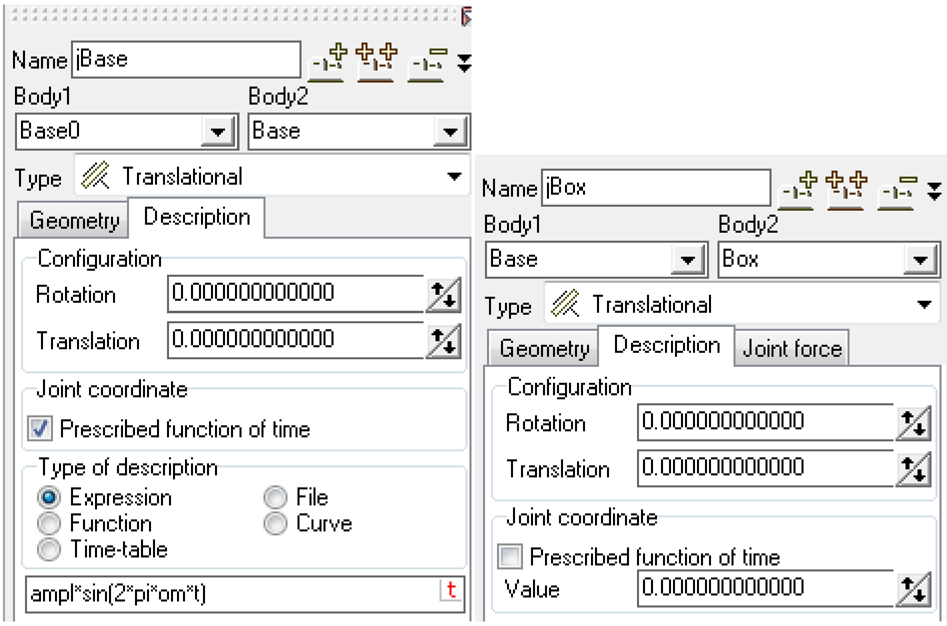
Figure 7.2. Damped oscillations for different values of damping ratio

7.3. Simple friction elements

Example: use and comparison of friction and elastic-friction elements (Chapter 2, Sect. *Force elements | Types of scalar forces*).



UM Model: LIBRARY\ElastFriction.



7.3.1. Bodies and joints

The model contains two bodies: *Box* and *Base*. Mass of the first body is parameterized by the identifier *mbox*, mass of the second body is not used because the second body motion is a prescribed function of time.

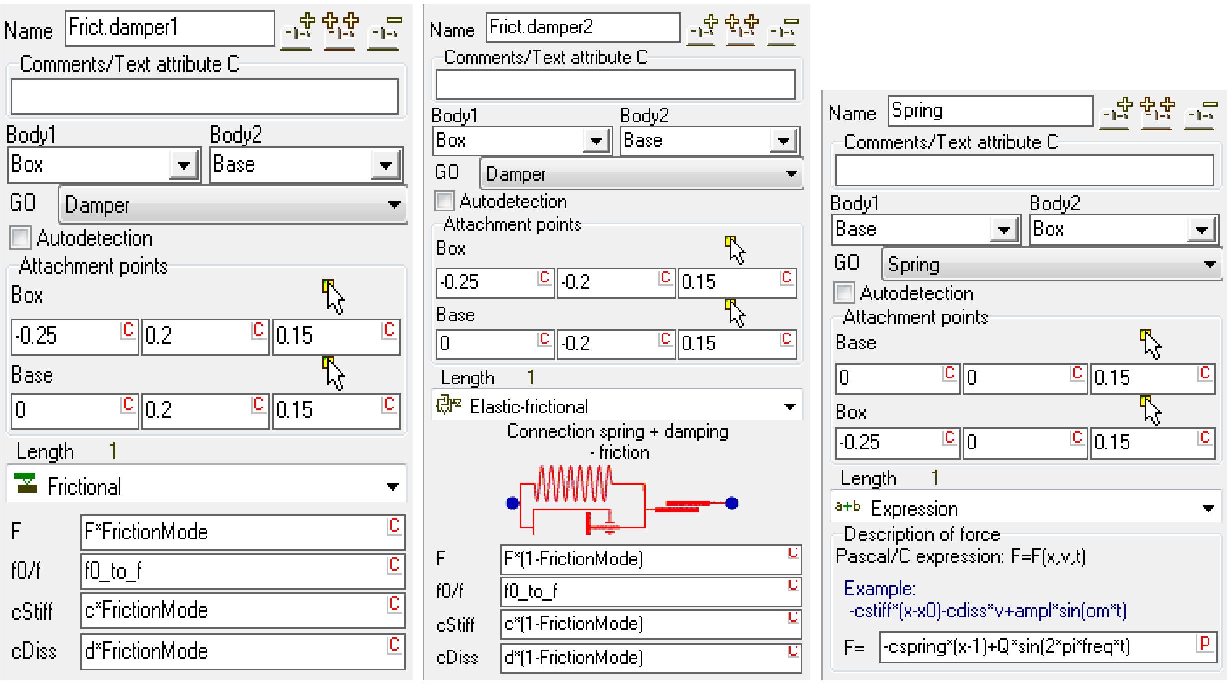
The *Box* body has one degree of freedom along the X-axis (joint *jBox*). A translational motion along the X-axis is set as harmonic oscillations described in the *jBase* joint with the help of the expression

$ampl*\sin(2*pi*om*t)$.

The expression is parameterized and allows changing both the amplitude (identifier *ampl*) and the frequency (identifier *om*). In particular, if *ampl*=0, the *Base* is fixed.

7.3.2. Force elements

The model contains tree bipolar force elements of different types connecting the *Box* and *Base* bodies. The elements oriented along the X-axis and produce forces depending on relative motions of the pair of bodies.



a) b) c)

Two of the tree elements define friction forces and model frictional dampers. The model is created for analysis and comparison of two types of friction, and one of these two elements will be switched off during simulations. The *FrictionMode* serves to the purpose of switching on/off the elements. If it is equal to 0, the *Frict.damper2* element is on, if it is 1, the *Frict.damper1* element is on.

The *FrictDamper1* element (Fig. a) corresponds to the *'Frictional'* type of a scalar force. The following parameters describe the element:

- The friction force is set by the expression

  F FrictionMode*,

  where *F* is the value of the force, and the factor *FrictionMode* switches on/off the force element whet its is equal to 1/0;
- Ratio of static to dynamic coefficients of friction i set by the identifier *f0_to_*, which default value is 1.2;
- Expression for stiffness at sticking is

  c FrictionMode*,

(*c* is the stiffness, and the factor *FrictionMode* switches it on/off);

- Expression for damping at sticking is

d FrictionMode,*

(*d* is the damping , and the factor *FrictionMode* switches it on/off).

The *FrictDamper2* element (Fig. b) corresponds to the ‘Elastic-friction’ type of a scalar force. It is parameterized by the same identifiers as the *FrictDamper2* element. The only difference consists in replacement of the factor *FrictionMode* by

1-FrictionMode,

i.e. the element is on by *FrictionMode* =1.

The last element *Spring* is set by the expression

-cspring(x-1)+Q*sin(2*pi*freq*t).*

The expression contains two summands. The first one *-cspring*(x-1)* corresponds to a linear elastic force with stiffness *cspring*. This part of the force can be switched off by *cspring*=0. If the stiffness is nonzero, the force vanishes at *x*=1.

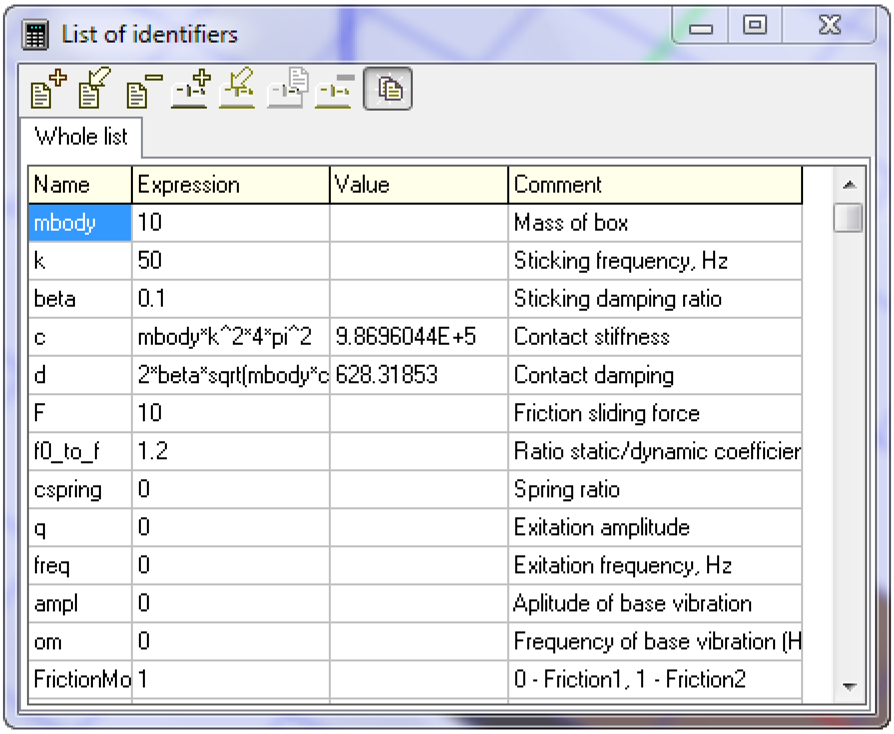
The second summand *Q*sin(2*pi*freq*t)* corresponds to a harmonic excitation. The excitation is off by *Q*=0. The excitation frequency is set by the identifier *freq*. It is clear, that the frequency unit is Hz, because pi is the standard identifier for $\pi = 3.1415926536$ ...

Though the element image is a spring, it can be

- switched off by Q=0 and *cspring*=0
- a linear elastic spring by Q=0 and *cspring* $\neq 0$
- a combination of elastic force and harmonic excitation by Q $\neq 0$, *cspring* $\neq 0$
- a pure harmonic excitation by Q $\neq 0$ and *cspring*=0

So, the image does not set the element properties, but it allows a visual identification of the element.

7.3.3. Identifiers-expressions and computation of stiffness and damping at sticking



Name	Expression	Value	Comment
mbody	10		Mass of box
k	50		Sticking frequency, Hz
beta	0.1		Sticking damping ratio
c	mbody*k^2*4*pi^2	9.8696044E+5	Contact stiffness
d	2*beta*sqrt(mbody*c	628.31853	Contact damping
F	10		Friction sliding force
f0_to_f	1.2		Ratio static/dynamic coefficier
cspring	0		Spring ratio
q	0		Exitation amplitude
freq	0		Exitation frequency, Hz
ampl	0		Aplitude of base vibration
om	0		Frequency of base vibration (H
FrictionMo	1		0 - Friction1, 1 - Friction2

The list of identifiers is shown the figure. It contains both known identifiers such as *c*, *d*, *FrictionMode*, and a number of additional identifiers: k is the contact frequency at sticking state if force elements, *beta* is the damping ratio at sticking. These two identifiers are used for evaluation of contact stiffness and damping parameters according to the following expressions (see Chapr.2, Sect. *Methodology of choice of contact parameters*):

$$c = 4\pi^2 k^2 m, \qquad d = 2\beta\sqrt{mc}.$$

Here we have an example of programming identifiers, and *c*, *d* are identifiers-expressions, which depend on other identifiers, located *above* in the list. As a result, values of these identifiers cannot be changed directly, but only by a modification of the frequency and damping ratio parameters.

7.3.4. Macro-command of identifiers

Before starting simulations of the model we consider how *macro-commands of identifiers* can be created. As a rule, *macro-commands* are used for simultaneous change of values of a group of identifiers.

The LIBRARY\ElastFriction model uses two macro-commands:

- **Type of friction** is the type of model of friction element. Its values are:

Standard – corresponds to the element of the frictional type (Frict.damper1). It sets FrictionMode=0;

Elastic – the elastic-friction model of force. Its sets FrictionMode=0.

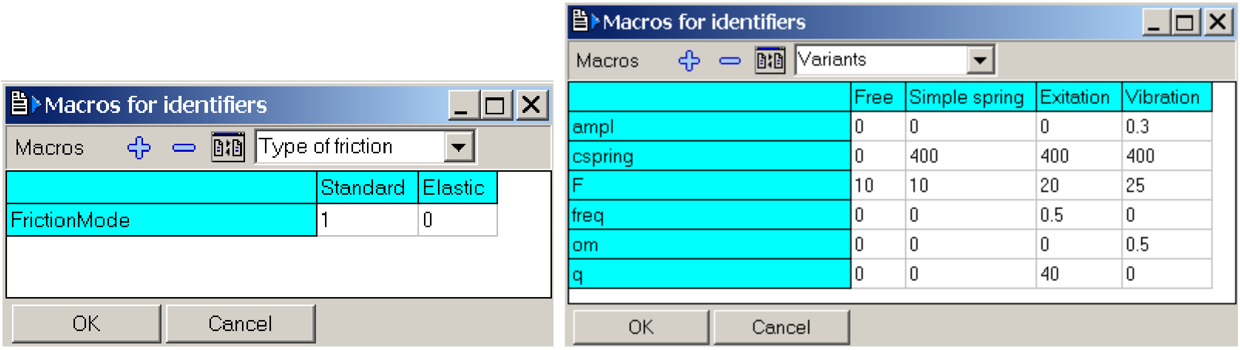
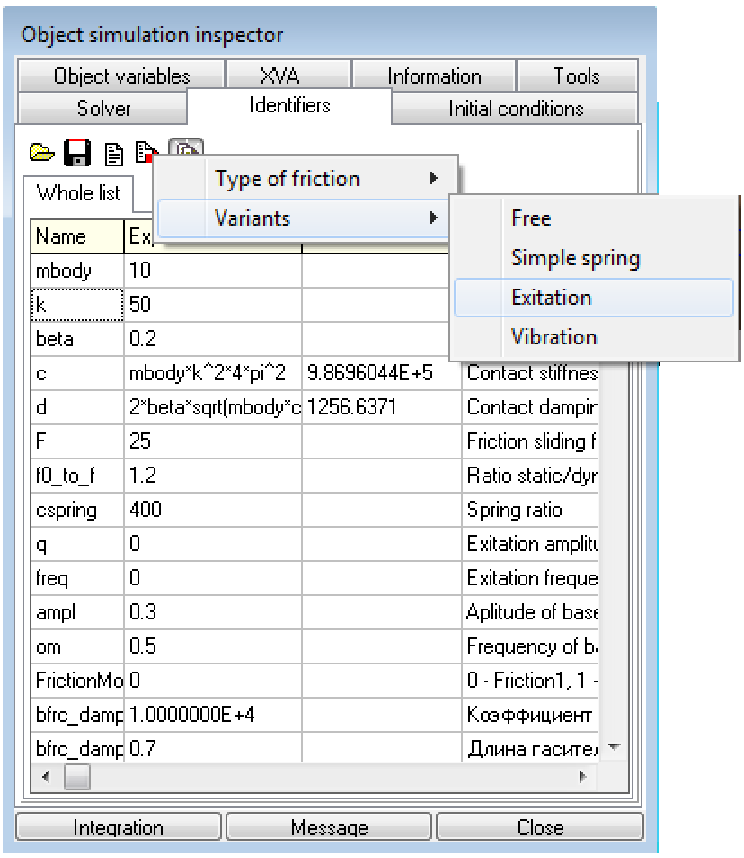


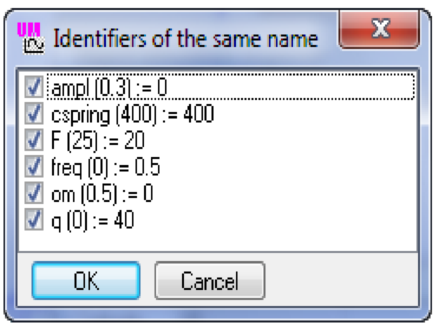
Figure 7.3. Macro-commands

- **Variants** is the assignment of values for a group of identifiers corresponding to different variants of simulations. Call of the macro-commands sets values of
    - amplitude and frequency of the base oscillations (*ampl*, *om*)
    - spring stiffness (*cspring*)
    - amplitude and frequency of force excitation (*q*, *freq*)
    - value of friction force (*F*)

7.3.5. Simulation

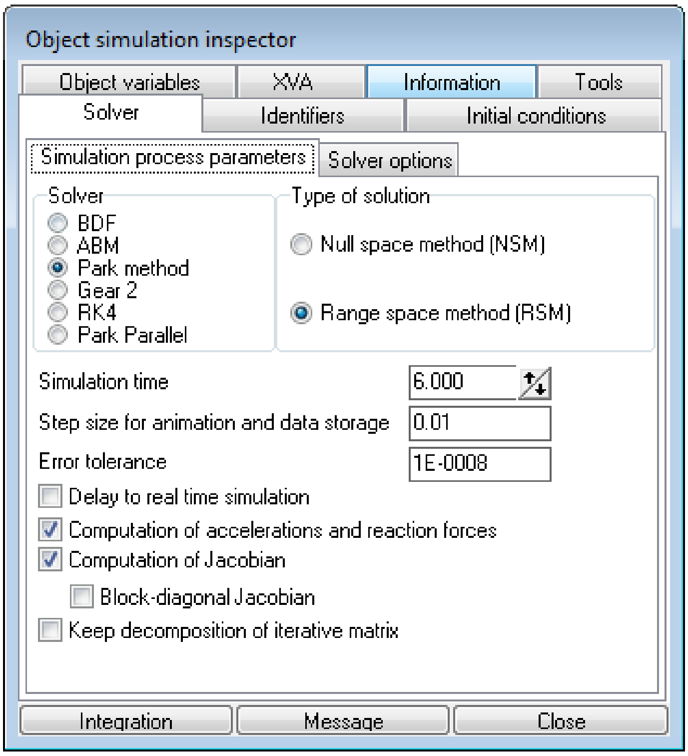
Different variants of simulation of the model can be set with the help of the macro-commands. To execute macros, open the Identifiers tab of the object simulation inspector, click the button end select either the *Type of friction* or the *Variants* item.





After clicking the item, a list with new values of the identifiers appears. To assign the new values, click the *OK* button.

Consider four variants of the object motion. It is recommended to use Jacobian matrices by the simulation, because the equations are stiff. The stiffness of the equations is caused both be the 'frictional' element at sticking mode and by the 'elastic-frictional' element at sticking and sliding. Recommended solver parameters can be found in the figure below.



Solver parameters

7.3.5.1. Free motion

Select the *Free* value of the *Variants* macros. The spring is off, and the box initial velocity is 1.5 m/s.

When the elastic-frictional element is on the plots shows dependences in Figure 7.4.

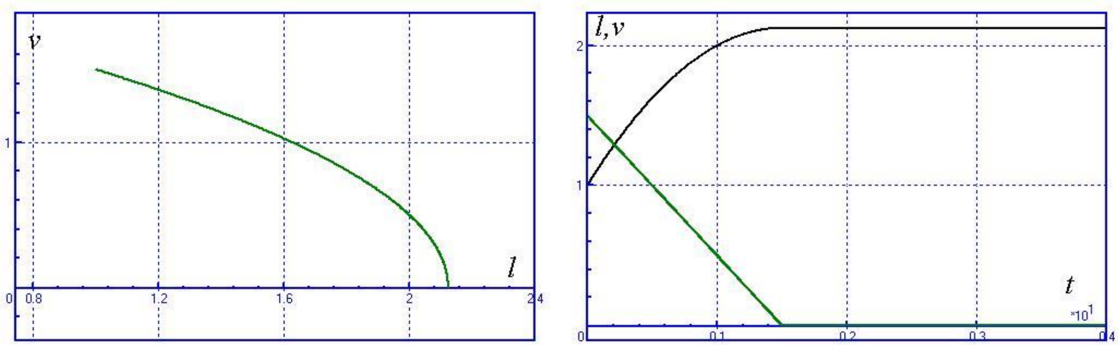


Figure 7.4. Box velocity versus lateral position (left); box velocity and lateral position versus time (right)

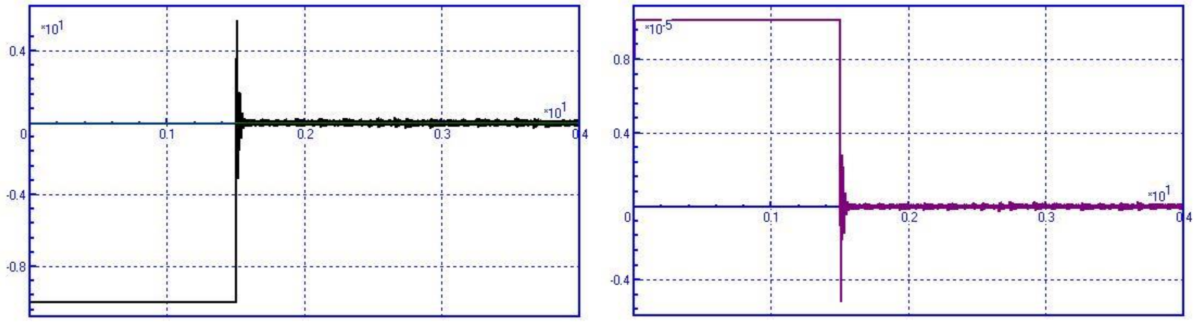
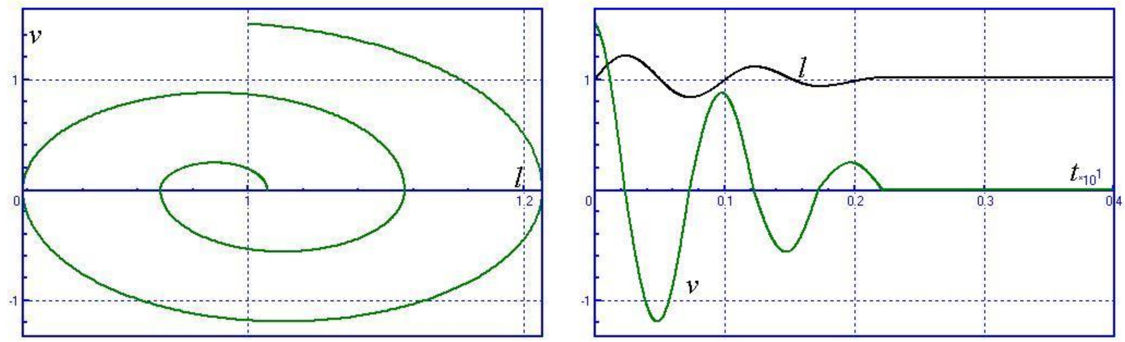


Figure 7.5. Friction force versus time (left); elastic deflection versus time (right)

For the ‘frictional’ element we have obtained quite the similar plots except the deflection, which is not available in an explicit form. Simultaneously we see the advantage of the ‘frictional’ model of the force versus the ‘elastic-frictional’ one: in the case of the ‘frictional’ model at sticking does not occur high frequent oscillation, which are seen in the case of ‘elastic-frictional’ model of the force. The oscillations amplitude decreases if the accuracy of integration increases (we have got the plots for the error tolerance 1.0×10^{-8}, which corresponds to a high precision of the simulation). These oscillations appear due to high stiffness of the elastic-friction force and due to approximate evaluation of Jacobian matrix of the force element. In contrary, the ‘frictional’ element realization is much more stable. Oscillations do not appear even for 1.0×10^{-6} error tolerance.

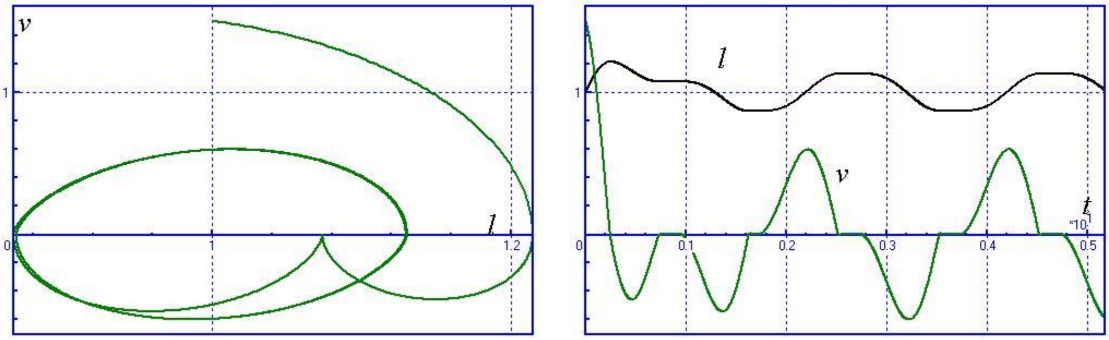
7.3.5.2. Influence of spring

Select the *Simple spring* value of the *Variants* macros. With the corresponding identifier values we have damped oscillations of the box. Damping is frictional. Figures show the velocity v versus the element length *l* as well as velocity and length in dependence on time.



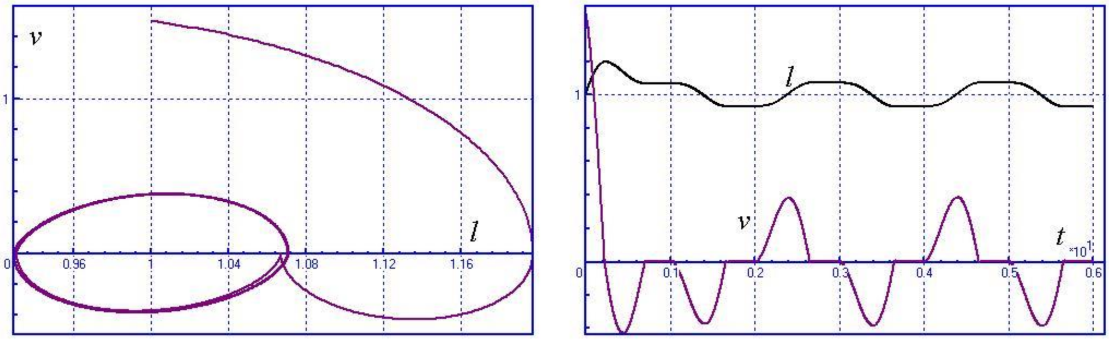
7.3.5.3. Excited oscillations

Select the *Excitation* value of the *Variants* macros. The excitation force differs now from zero.



7.3.5.4. Excited oscillations due to base vibration

Select the *Vibration* value of the *Variants* macros. The base vibrates according to the harmonic law.



7.4. Elasto-friction element 2

Example: illustration to the elasto-friction 2 (Chapter 2, Sect. *Elasto-friction force* 2).

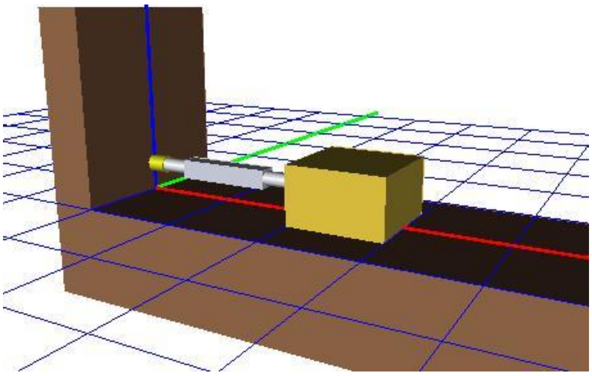


Figure 7.6. Model of free vibrating body

UM Model: LIBRARY\ElastFriction2.

7.4.1. Model description

The model in Figure 7.6 includes of a body *Box* with one degree of freedom along the X-axis of SC0, a bipolar force element of the elastic-friction 2 type with the name – *Friction-Elastic element*, and a fore element of general type with one projection on the X-axis with the name *Constant Force*.

The following identifiers are used for parameterization of the model:

mbody – mass of body;

cspring – stiffness of the elastic-friction element in series with friction;

cspring2 – stiffness of the elastic-friction element in parallel with friction;

ffr – dynamic coefficient of friction;

ffr0 – static coefficient of friction;

L0 – length of the elastic-friction element in an undeformed state (*L*0=1 corresponds to the element length for zero value of the coordinate);

Fx – constant force acting on the body along the X-axis.

See Figure 7.7 for default values of identifiers.

Name	Expression	Value	Comment
mbody	10		Mass of box
cspring	400		Serial spring ratio
cspring2	100		Parallel spring ratio
ffr0	0.2		Static coeff. of friction
ffr	0.2		Dinamic coeff. of friction
l0	1		Length of undeformed element
Fx	0		Longitudinal constant force

Figure 7.7. Default values of identifiers

7.4.2. Properties of elastic-friction 2 in frequency domain

To get properties of elastic-friction 2 in frequency domain, open the model in the Simulation module. Use the **Tools |Force analysis** menu item or the *Ctrl+F* hot key to open the window *Analysis of force elements* (Figure 7.8).

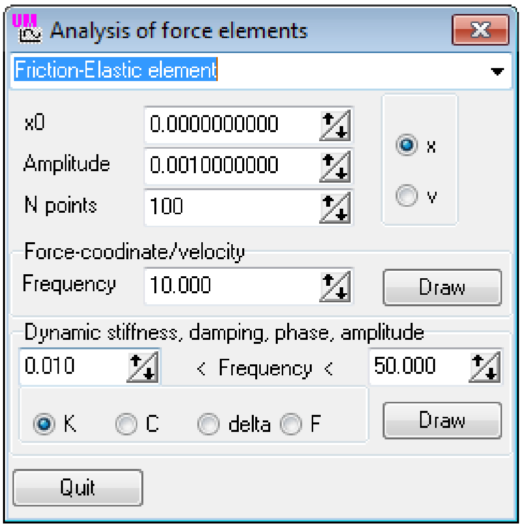


Figure 7.8. Window for analysis of forces in frequency domain

This tool computes a force element response on the following harmonic excitation

$$x = \alpha \sin(2\pi f t),$$

where *f* is the frequency in Hz. Use the *Draw* buttons to obtain plots

- Dependences *f(x)*, *f(v)* for fixed frequency of excitation

$$\left(v = \dot{x} = 2\pi f a cos(2\pi f t)\right);$$

- Dynamic stiffness (K), phase (delta) of the element as well as force amplitude (F) versus frequency in a given interval

A hysteresis for the elastic-friction 2 force versus a displacement $x` = x - L_0$ is shown in Figure 7.9 for tree values of the amplitude of the excitation 1,2,3 mm. The plots do not depend on the frequency.

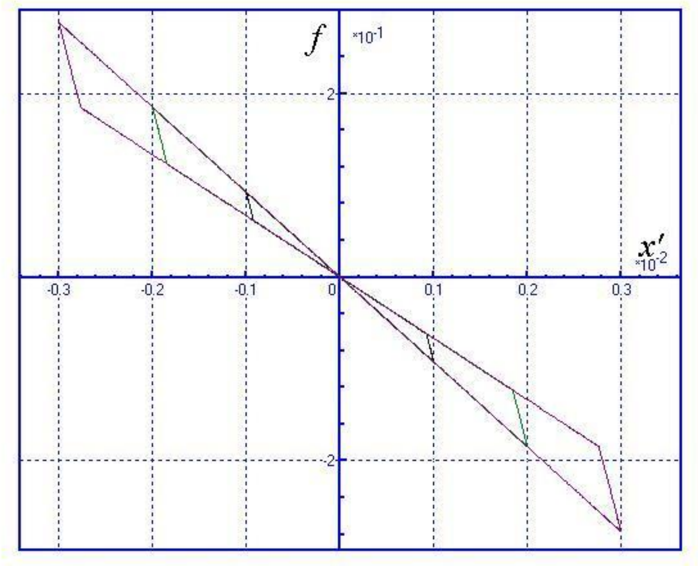


Figure 7.9. Force versus displacement

Dynamic stiffness of the element does not depend on the frequency, but increases with the coefficient of friction, Figure 7.10.

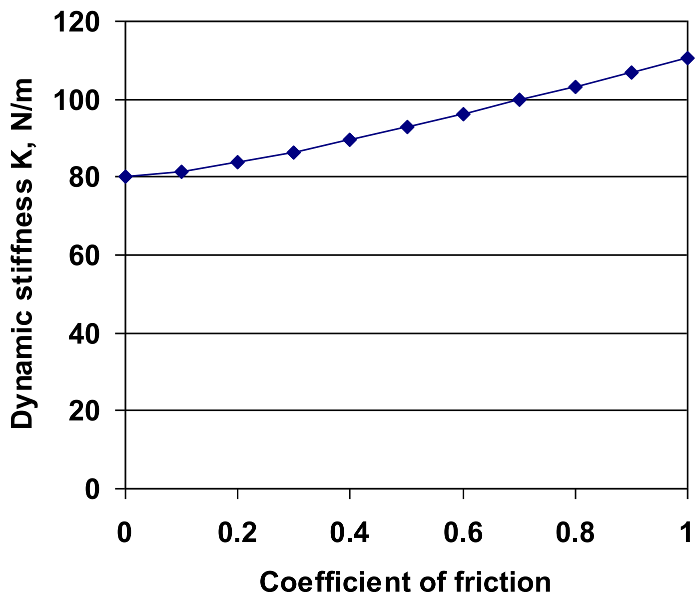


Figure 7.10. Dynamic stiffness versus coefficient of friction

Let us consider how we have got the plot in Figure 7.10. Select the dynamic stiffness K in the radio group. Set sequence and equal values 0, 0.1, 0.2… for the identifiers of static and dynamic coefficients of friction, and click the *Draw* button in the *Dynamic stiffness, phase, amplitude* group to get a hysteresis. Use the mouse within the graphic window to get the numeric value of the stiffness.

Figure 7.11 shows the delay of the response (phase) versus the friction coefficient. This parameter characterizes damping properties of the element.

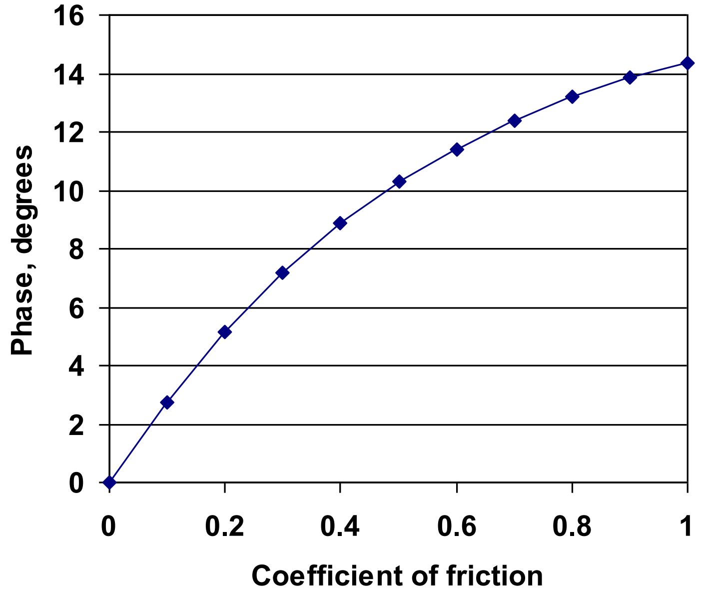


Figure 7.11. Phase versus coefficient of friction

7.4.3. Simulation results

Run the simulation for default value of identifiers (Figure 7.7) and for initial positive velocity of the body 2m/s. The force versus the length of element hysteresis time is shown in Figure 7.12 (left). Dependences ‘full element deflection versus time’ and ‘deflection of spring in parallel with friction x_2 versus time’ are compared in Figure 7.13. An increased fragment of the figure shows a sticking mode where the deflection x_2 is a constant.

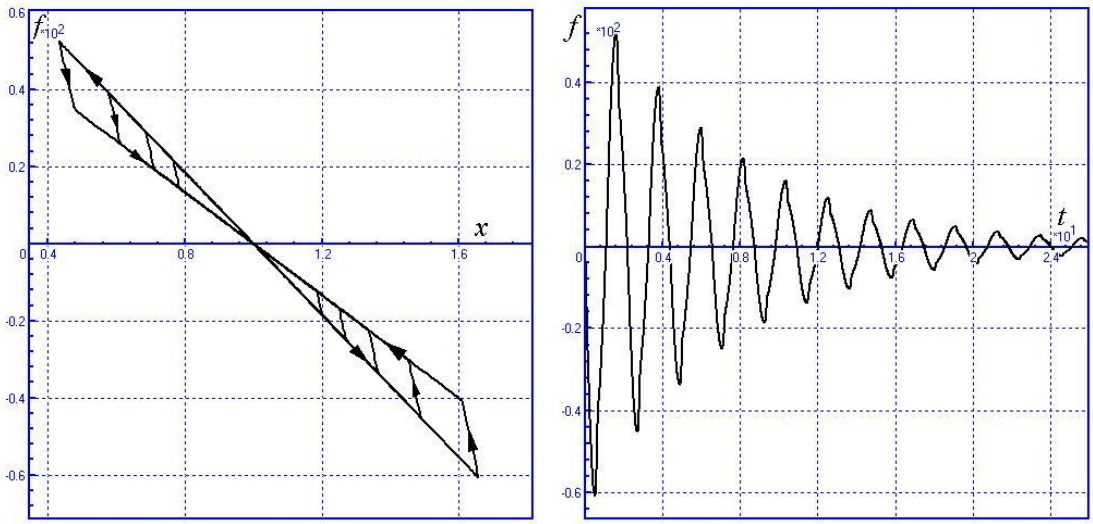


Figure 7.12. Force versus element length and time

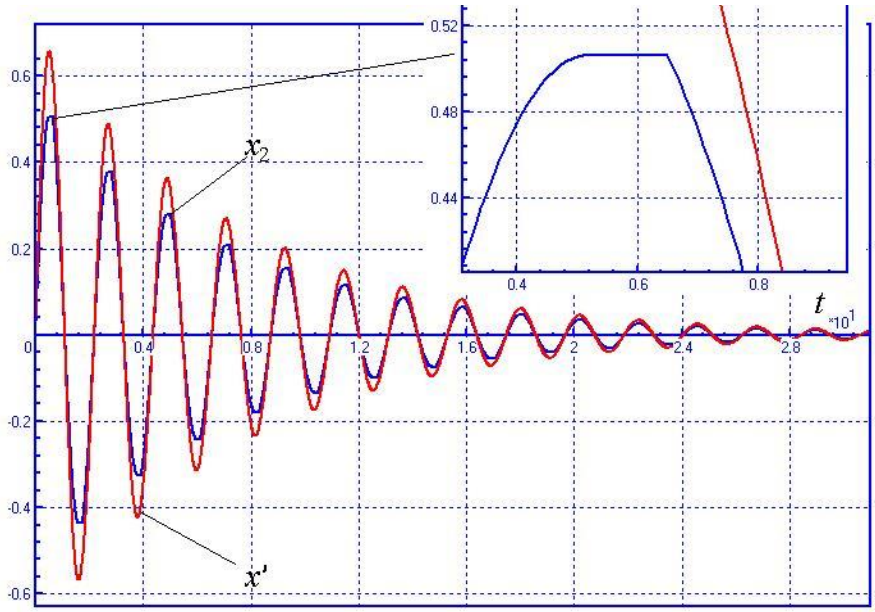
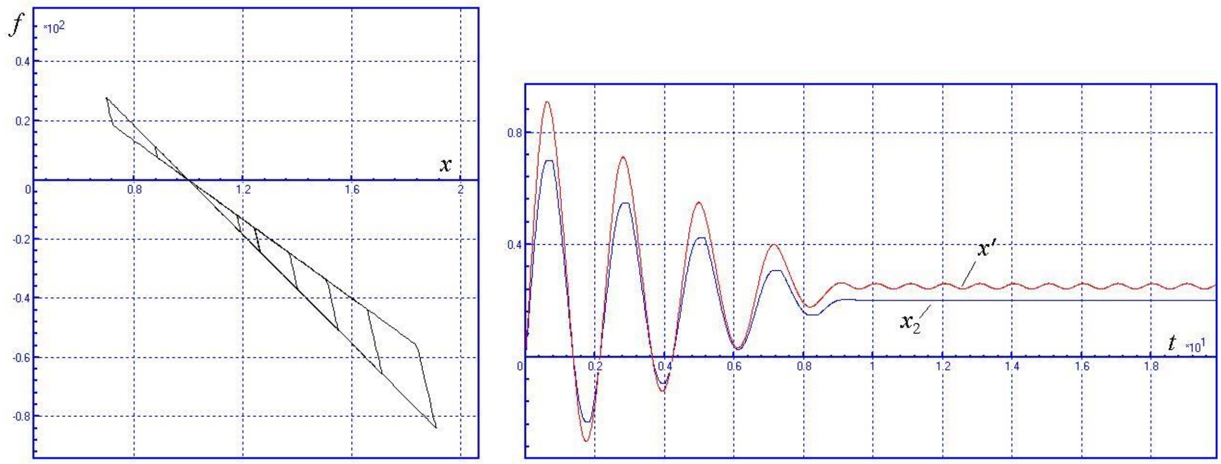


Figure 7.13. Full element deflection $x` = x - L_0$ and spring in parallel to friction deflection x_2 versus time

Now set the nonzero value 20N to the constant force F_x. The right figure shows that at sticking the oscillations are undamped.



Make a number of additional simulations. Make sure that no damping presented for zero value of coefficient of friction.

7.5. Convel joint

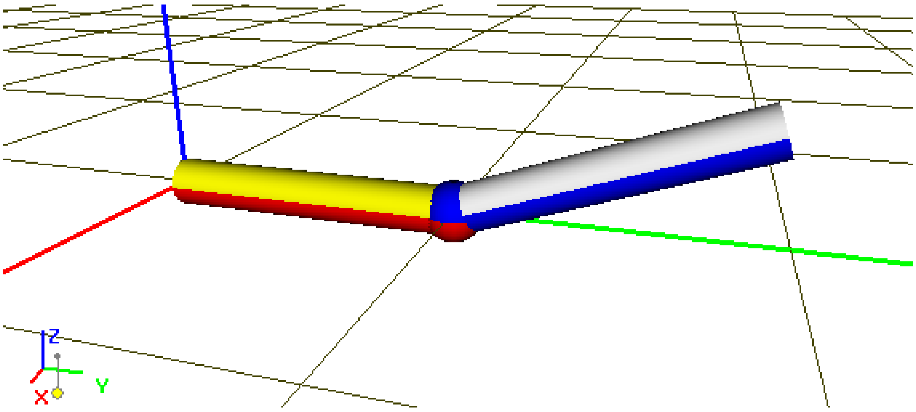


Figure 7.14. Model: general view

Example in Figure 7.14 illustrates two variants of development of model with usage of **convel** (constant velocity) **joint**. The joint theoretical background can be found in Chapter 2, Sect. **Joints | Description of joints | Convel joint**.

In this example we consider two models, realized in different ways, but solving the save problem: development of a model in which two shafts have equal angular velocity even when one of them is inclined relative to another one.

UM Models: LIBRARY\Convel1; LIBRARY\Convel2.

7.5.1. Description of models

Both models contain two bodies modeling shafts, Body1 and Body2.

One rotational degree of freedom relative to Base0 is specified for body Body1 by the joint jBody1. A joint torque described by the expression

*F0*exp(-mu*t)*

brings shafts into rotation. Angular velocity exponentially tends to a definite constant value.

An joint introducing degrees of freedom the Body2 relative to Base0 is realized in the models by different ways.

- Model Convel1

Joint jBody2 similar to jBody2 introduces on rotational degree of freedom of Body2 relative to Base0. To parameterize the inclination of the second shaft relative to the first one, the joint of the 'generalized type' is used, which contains the following sequence of elementary transformations (see Chapter 2, Sect. **Joints | Description of joints | Generalized joint**):

$$T_1 = \{tc, e = (0,0.5,0)\},$$
$$T_2 = \{rt, e = (1,0,0), s = alpha * pi/180\},$$
$$T_3 = \{rv, e = (0,1,0)\},$$

The second transformation is of type rt instead of rc to set the shaft inclination by an identifier. Thus, the identifier alpha sets this angles in degrees. Note that elementary transformation of the rc allows numeric value of angle only.

- Model Convel2

Unlike the Convel1 model, in this case the jBody2 introduces 6 degrees of freedom, i.e. it does not limits motion of Body2 relative to the Base0. Due to the set of elementary transformations listed below, the first rotational degree of freedom corresponds to rotation of the second shaft about the inclined axis, which position is set similar to the Convel1 model.

$$T_1 = \{tc, e = (0,0.5,0)\},$$
$$T_2 = \{rt, e = (1,0,0), s = alpha * pi/180\},$$
$$T_3 = \{tv, e = (1,0,0)\},$$
$$T_4 = \{tv, e = (0,1,0)\}$$
$$T_5 = \{tv, e = (0,0,1)\}$$
$$T_5 = \{rv, e = (0,1,0)\}$$
$$T_6 = \{rv, e = (1,0,0)\}$$
$$T_7 = \{rv, e = (0,0,1)\}$$

The jBody1_Body2 joint of the type 'CV joint' connecting the bodies Body1 and Body2 is presented in both models. This joint allows transferring an angular velocity to the second shaft equal to the rotational speed of the first shaft.

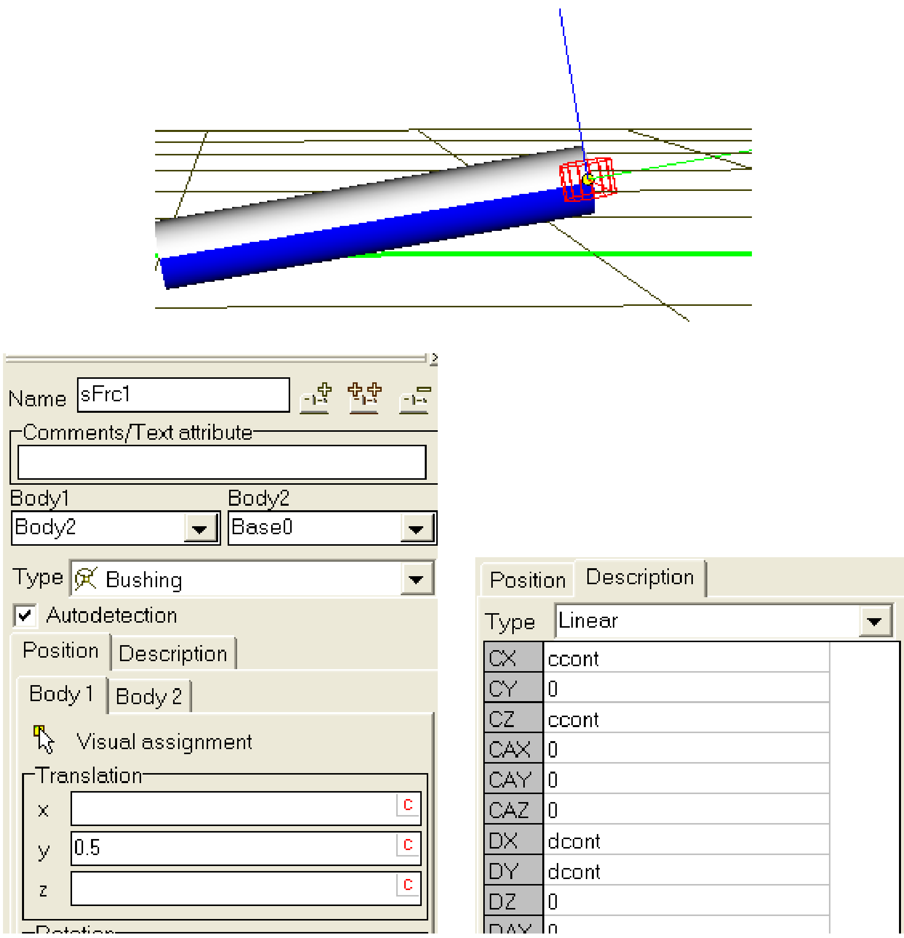


Figure 7.15. Force element of 'bushing' type

The description of the model Convel1 is finished unlike the Convel1 model, which is still not ready. The second shaft in the Convel2 model has two degrees of freedom relative to the first shaft because the CV joint takes away four degrees of freedom (three translational and one rotational). To avoid the shaft motion along the redundant degree of freedom, a special force element of the ‘bushing’ type is added in the model Convel2, Figure 7.15.

It is necessary to draw attention to the following features of this element description. Body2 is assigned as the **first** body and Base0 as the second one. This sequence of assignment has several advantages. First, if the **Autodetection** is checked, it is enough to enter the position of the force element in SC of the body Body2 only, (0, 0.5, 0). Attachment point for the second body, which in particular depends on the shaft inclination angle, will be computed by the program automatically. Second, bushing axes correspond exactly to axes *x* and *z* of Body2-fixed system of coordinates.

Thus, this force element introduces an elastic bushing, which prevents the shaft travel perpendicular to its axis, and it fact it allows the shaft rotation about the inclined axis if symmetry.

The main advantage of the Convel2 in comparison with the Convel1 is its **static certainty**. The model Convel1 is **statically uncertain** due to redundant constraint imposed by the convel joint.

7.5.2. Simulation of shafts

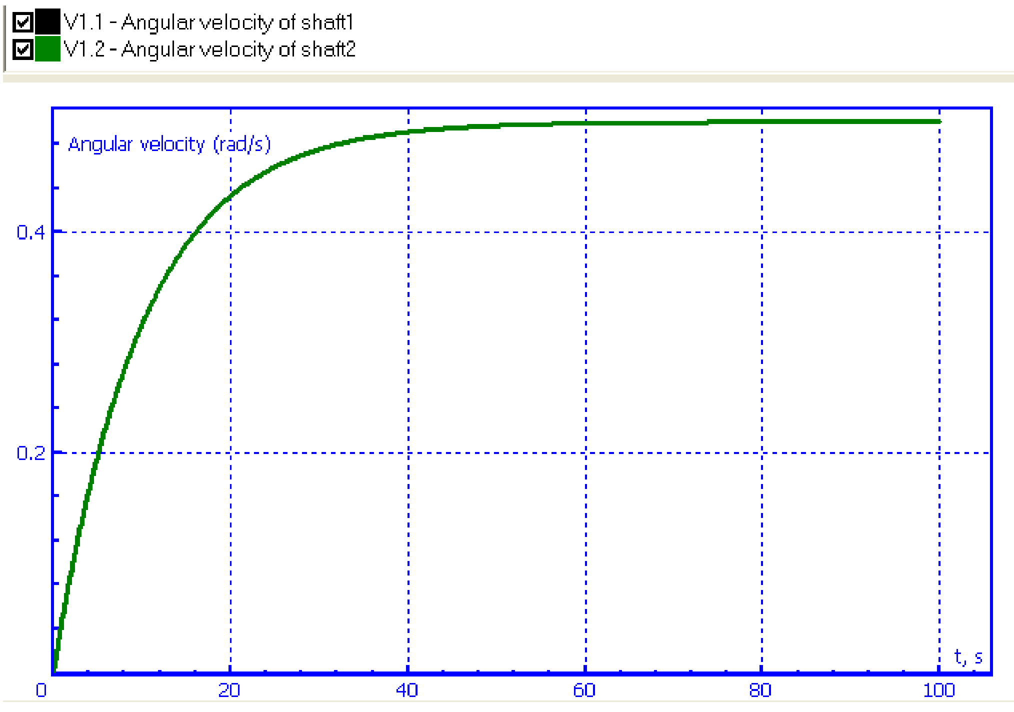


Figure 7.16. Shaft angular velocities vs tine

By simulations the models shows quite similar results for angles of rotation, angular velocities and accelerations of the shafts. Angular variables are exactly equal for both shafts in model Convel1, Figure 7.16, and slightly different for model Convel2 due to small dynamic deflections in the bushing.

Name	Expression	Value	Comment
alpha	10		
F0	0.01		
mu	0.1		
ccont	0		
dcont	0		
m	10		
ix	5		
iy	0.1		

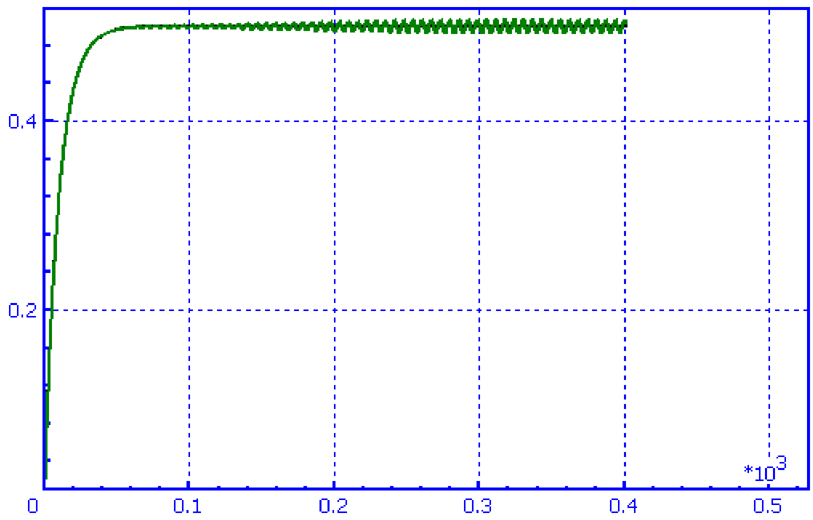


Figure 7.17. Angular velocities vs time by removed bushing

To understand the influence of the elastic bushing in the model Convel2, set zero values for identifiers *ccont*, *dcont* and increase the duration time to 500s. In this simulation the second shaft does not keep the inclination angle, and angular velocities of shafts differ considerably, Figure 7.17.

7.6. Cube on a plane

Model (see Figure 7.18) shows as it is possible to describe and use ‘Plane-Points’ contact interaction. Free body 1 with 6 d.o.f. falls down to planes 2 and 3. There is a quaternion joint between body 1 and base. Planes 2 and 3 belong to the base. Contact interaction between body 1 and planes is described with the help of two ‘Plane-Points’ contact forces.

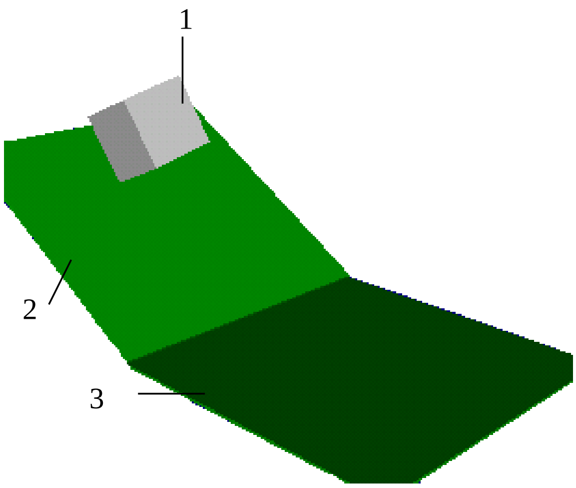


Figure 7.18. Body on a plane

UM Model: LIBRARY\Falling.

7.6.1. Model features

7.6.1.1. Description of contact interaction

Contact interaction is described with the help of two ‘Plane-Points’ contact forces. Contact points belong to body 1 and set in the its vertexes. Contact planes are described with the help of any point which is belong to the plane and the vector of external normal. In the Table 7.1 points and vectors for the both planes are given.

Table 7.1

Description of the contact planes

Contact plane	Point on the plane	External normal
2	[0; 0; 0]	[0; 0.5; 0.86]
3	[0; 0; 0]	[0; 0; 1]

7.6.1.2. Quaternion joint

Body 1 has 6 d.o.f. which are introduced with the help of quaternion joint. It is recommended to use quaternion joint if body might has arbitrary orientation as body 1 in this example because quaternion joint has no degeneration orientation dislike other multi-dimensional joints. But on the other hand using quaternion joint takes us extra computational efforts.

7.7. Jumping body

Quaternion joint connects body 1 and the base and has six d.o.f. (see Figure 7.19). Translation joint connects body 2 and body 1. There is a viscous-elastic force in the translation joint. There are three ‘Plane-Circle’ contact forces, which act between plane 3 and bottom side of body 2, and the both sides of body 1.

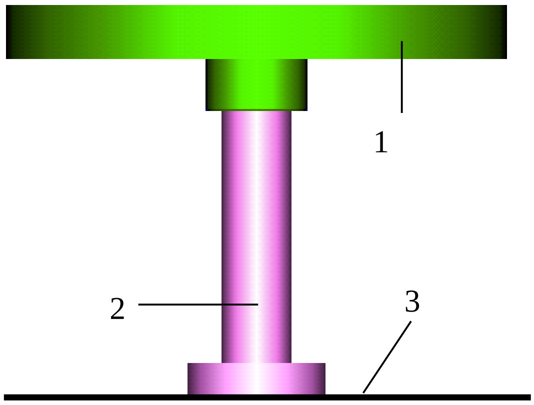


Figure 7.19. Jumping body

UM Model: LIBRARY\Fidget.

7.7.1. Model features

7.7.1.1. Graphic object of body 1

To form graphic object of body 1 the profile graphic element is used: Type of section – Curve 2D, Axis curve – Circle. It is necessary to set a section in the bottom half-plane as it is shown at Figure 7.20.

7.7.1.2. Description of the joint force

There is an elastic-damping force in the translation joint. Type of joint force is expression, very expression: -c*x-d*v, where c - stiffness coefficient, d –damping coefficient.

7.7.1.3. Description of contact forces

Three ‘Plane-Circle’ contact forces are in introduced in model as well. Pay attention that it is necessary to set the external normal, but it is possible to set any of normals.

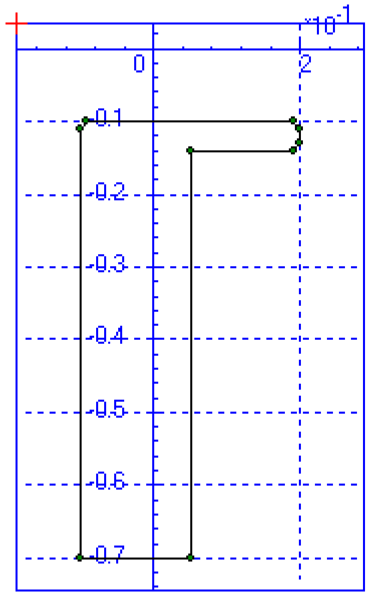


Figure 7.20. Section of profile object (body 1)

7.8. Simple oscillations

Model shows six various ways to describe simple spring. The same masses are assigned for all of bodies and the same stiffness and initial deviations are assigned for springs. So, all bodies oscillate equally. All bodies connect to the base with the help of translation joint. Detailed data about used force elements are given in the Table 7.2.

UM Model: LIBRARY\Forces

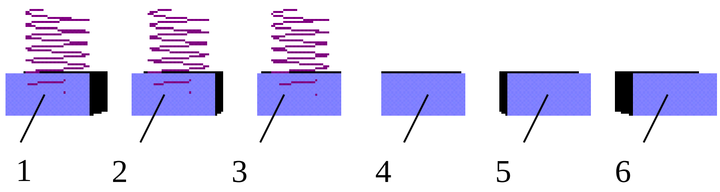


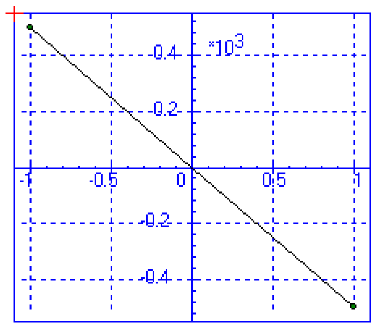
Figure 7.21. General view of the model

Table 7.2

Types of force elements

Number of body	Type of force	Subtype of force	Comment
1	Bipolar	Linear	c=cstiff; x0=0
2	Bipolar	Points	see Figure 7.22
3	Bipolar	Expression	F=-cstiff*x
4	Joint	Linear	c=cstiff; x0=0
5	Joint	Points	see Figure 7.22
6	Joint	Expression	F=-cstiff*x

Stiffness of springs (*cstiff* identifier) is 500 Nm. In the case of *Points* subtype of forces a graphic characteristic is described like it is shown in Figure 7.22.



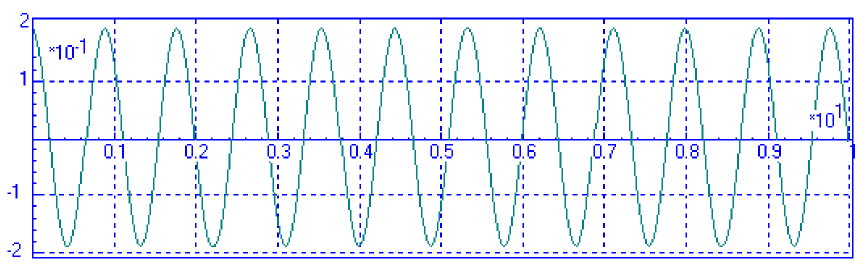


Figure 7.22. Graphic description of force (left)

Figure 7.23. Oscillations (right)

7.8.1. Model features

7.8.1.1. Difference of using bipolar and joint forces

Elastic forces for bodies 1, 2, 3 are described as bipolar forces with spring graphic objects. Elastic forces for bodies 3, 4, 5 are described as joint forces. Joint force cannot be assigned with graphic object.

7.8.2. Simulation results

If masses of all of bodies are 10 kg then they oscillate like it is show in the Figure 7.23.

7.9. Three bodies on springs

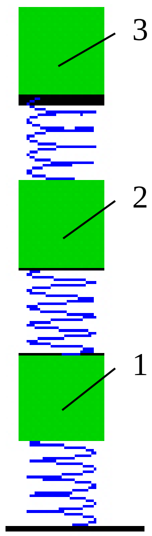
The model consists of three bodies linked with springs. Every body has one degree of freedom – translation along axis *z*. The spring is realized by means of bipolar forces which link the bodies (body 1 is linked with *Base0*).

UM Model: LIBRARY\Frequencies.

7.9.1. Model features

7.9.1.1. Description of graphical images of springs

Use graphical element *spiral* for definition of graphical image of spring. Take *Height* of spiral equal to 1, it means that height of spiral is the product of distance between attachment points of force element into 1. It visualizes the process of deflection of spring.



	Re	Im
1	17.6613	
2	43.8189	
3	67.2512	

Figure 7.24. Bodies on springs (left)

Figure 7.25. Natural frequencies of the system (right)

7.9.1.2. Computing of natural frequencies of the system

Use Linear analysis for computing of natural frequencies of the system. Oscillation frequencies and modes are represented on tab Frequencies (Figure 7.25). If obtained frequencies compare with peaks on graphs of power spectral density of oscillation of any of the bodies then obviously that peaks correspond with natural frequencies (Figure 7.26).

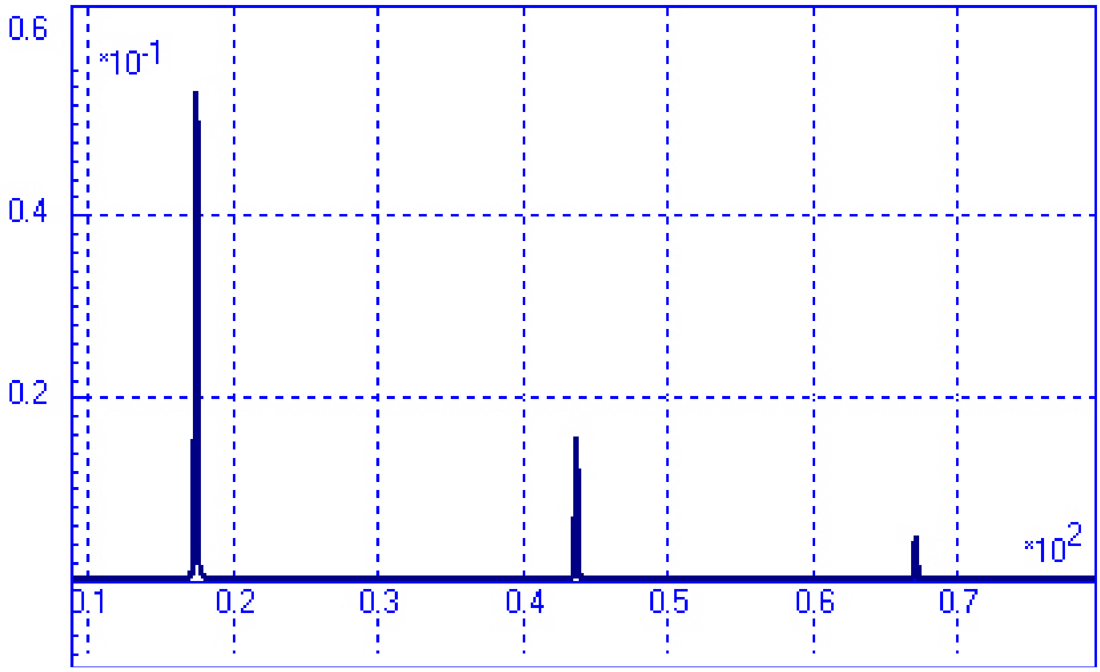


Figure 7.26. Power spectral density of oscillation of body 3

7.10. Body on rough plane. Self-excited oscillations

Model (see Figure 7.27) shows a movement of a body on a rough plane. Body 2 moves on plane 1 under the influence of spring 4 which is connected to body 5. Body 5 moves with the constant velocity along guide 3.

UM Model: LIBRARY\Oscillations.

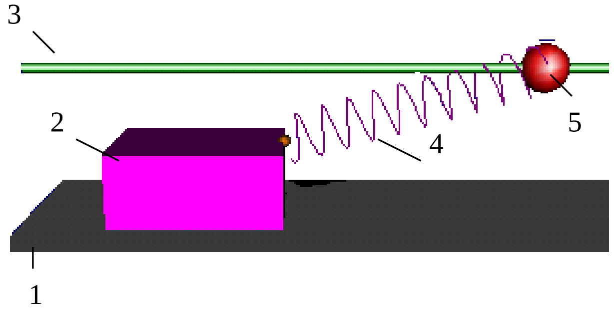


Figure 7.27. Model: general view

7.10.1. Model features

Contact interaction

Contact interaction between body 2 and plane 1 is described with the help of ‘Points-Plane’ contact force. Contact points belong to body 2 and they are set in its bottom vertexes. The contact plane belong to the base, external normal vector is N={0; 0; 1}.

Preset movement

The movement of body 5 along the guide 3 is described with the help of translation joint. And joint coordinate in it is set as a time function. In this example it is *‘v_leader*t’*, where *v_leader* is the velocity of body 5. In such a way preset movement is described.

Graphic object of the base

Pay attention, that graphic object of the base consists of plane 1 and guide 3 in this example.

7.10.2. Results

Self-excited oscillations arise in this example. Graphs of position and velocity of body 2 when body 5 moves with constant velocity along guide 3 are given in Figure 7.28.

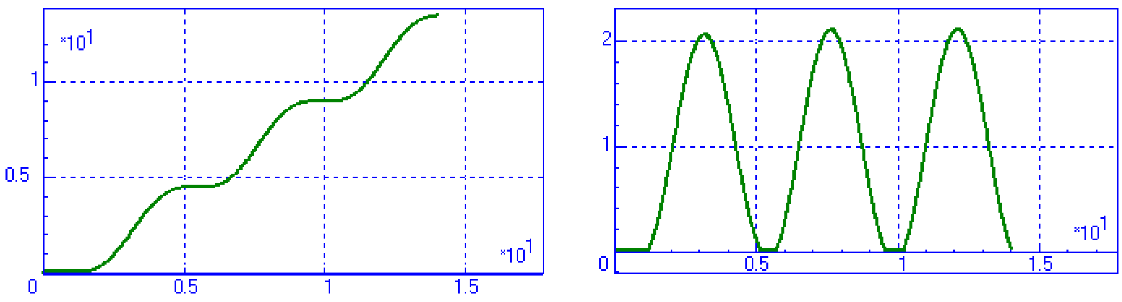


Figure 7.28. Time histories of position and velocity of the body 2

7.11. Beam on rollers

The model is the beam lying on two whirling rollers (Figure 7.29). Beam is body with 3 degrees of freedom (two translations along axes *y* and *z* and rotation around axis *x*). Rollers revolve on axis x, angle of revolving is explicit time function.

UM Model: LIBRARY\Rollers.

Figure 7.29. General view of the model

7.11.1. Model features

7.11.1.1. Definition of contact interaction “Circle-Plane”

Rollers interact with beam by means of contact forces. Define parameters and geometry of contact interaction on tab *“Contact forces”* to set the forces. In present model use type of contact *“Circle-Plane”*. A plane is defined with point and normal to plane, circle is determined with coordinates of center, radius and normal to circle.

7.11.1.2. Definition of graphic image as Archimedean spiral

Polar equation of the Archimedean spiral is:

$$\rho = \alpha\varphi.$$

Use conversion formula from polar coordinates to Cartesian for plotting spiral in plane yz:

$$y = \rho cos\varphi,$$
$$z = \rho sin\varphi.$$

If p1 is defined as φ (see Figure 7.30) then equation of Archimedean spiral is:

$$y = a \cdot p1 \cdot \cos p1,$$
$$z = a \cdot p1 \cdot \sin p1.$$

If p1 is changed from 0 to π then for “twisted” at $b\pi$ radians spiral and $\rho(\pi) = r$ equations are:

$$y = r \cdot \frac{p1}{\pi} \cdot \cos(b \cdot p1),$$
$$z = r \cdot \frac{p1}{\pi} \cdot \sin(b \cdot p1).$$

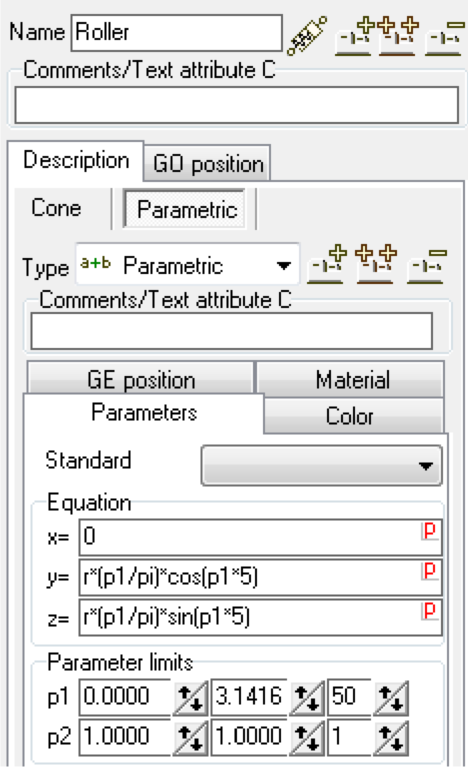


Figure 7.30. Definition of parametric graphic image

The result of variable friction forces is beam oscillates. The oscillation of center of mass is represented on Figure 7.31.

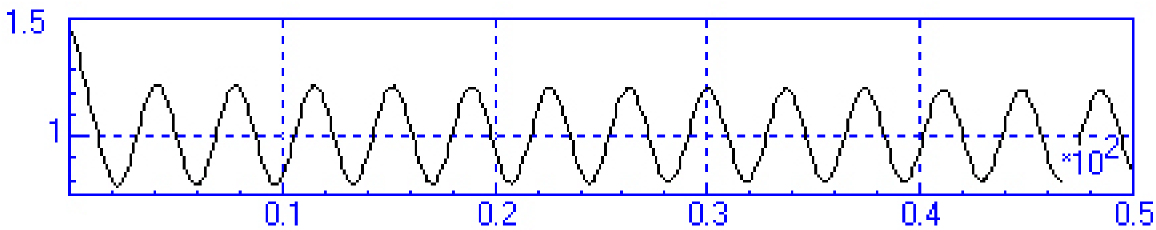


Figure 7.31. Oscillations of beam center of mass

7.12. Preset movement

Model shows how to describe movement as a time function with the help of generalized joint. Body 3 moves in a circle 1. Pendulum 2 is connected to body 3 with the help of rotational joint. Movement of body 3 is preset and described as a time function.

UM Model: LIBRARY\Round.

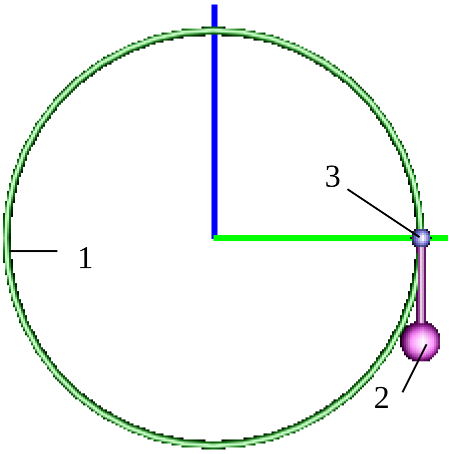


Figure 7.32. General view of the model

7.12.1. Model features

Description of movement of body 3 relative to base

The movement of body 3 is described with the help of generalized joint. There are two elementary transformations in the generalized joint. Detailed information about these elementary transformations (ET) is given in Table 7.3.

Table 7.3

Elementary transformations in generalized joint

ET index	ET type	ET vector	Expression
1	tt	Y-axis (0; 1; 0)	r*cos(omega*t+phase)
2	tt	Z-axis (0; 0; 1)	r*sin(omega*t+phase)

Table 7.4

Identifiers and values

Identifiers	Values	Comments
r	2	Radii of circle 1
omega	1	Angular frequency of movement of body 3
phase	0	Phase

Graphic object of base

Pay attention that graphic object of the base consists of circle 1 in this example.

7.12.2. Results

Time history of angle of turning of pendulum is given in the Figure 7.33.

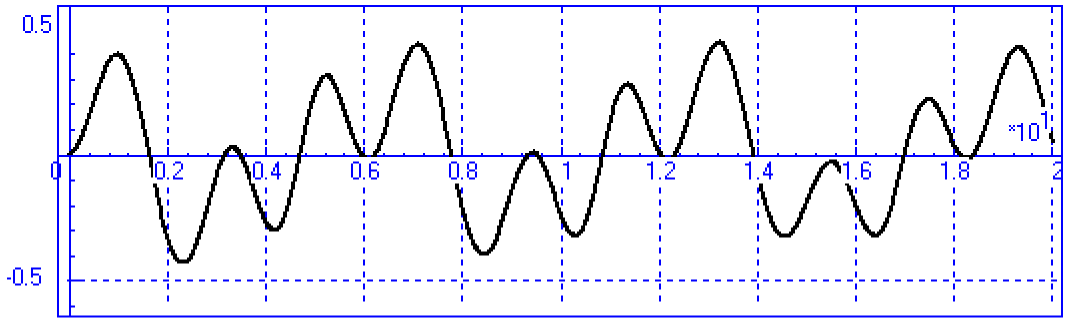


Figure 7.33. Angle of turning of pendulum vs. time

7.13. Use of generalized joint

Model shows usage of generalized joint. A slider-crank mechanism is shown in Figure 7.34. There are three bodies: crank 1, con-rod 2, slider 3. And there are four generalized joint: 1, 2, 3 with rotational d.o.f. and 4 with translation d.o.f.

UM Model: LIBRARY\Slider.

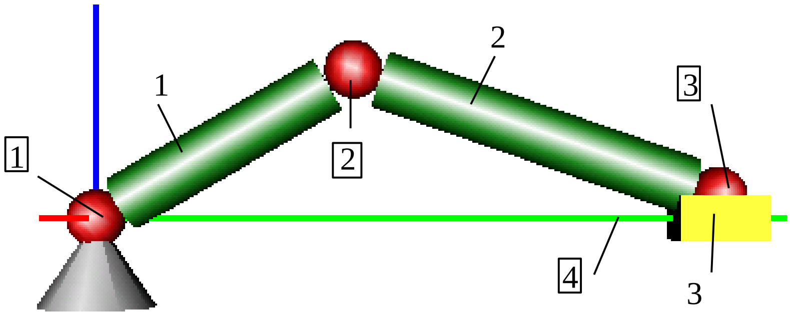


Figure 7.34. Slider-crank mechanism

7.13.1. Description of generalized joint

All joints in the model are described as generalized. Detailed information about elementary transformation (ET) in joints is given in the Table 7.5. Detailed information about generalized joint you can find in *User's guide*, paragraphs 2.3.4, 3.4.7.5.

Table 7.5

Elementary transformations in generalized joint

Joint index	ET index	ET type	ET vector	ET parameter
1	1	rv	Axis X (1; 0; 0)	-
	2	tc	Axis Z (0; 0; 1)	-a*0.5+r
2	1	tc	Axis Z (0; 0; 1)	-a*0.5-r
	2	rv	Axis X (1; 0; 0)	-
	3	tc	Axis Z (0; 0; 1)	-b*0.5+r
3	1	tc	Axis Z (0; 0; 1)	-b*0.5-r
	2	rv	Axis X (1; 0; 0)	-
	3	tc	Axis Z (0; 0; 1)	-d*0.25
4	1	tc	Axis Y (0; 1; 0)	c
	2	tv	Axis Y (0; 1; 0)	-
	3	tc	Axis Y (0; 1; 0)	d*0.5

7.14. Simple pursuit game

The model is an implementation of a simple mathematical pursuit game (Figure 7.35), in which a target 1 moves in a certain path (circle) and a pursuer 2 must catch it using some pursuit strategy.

UM Model: LIBRARY\MathGame.

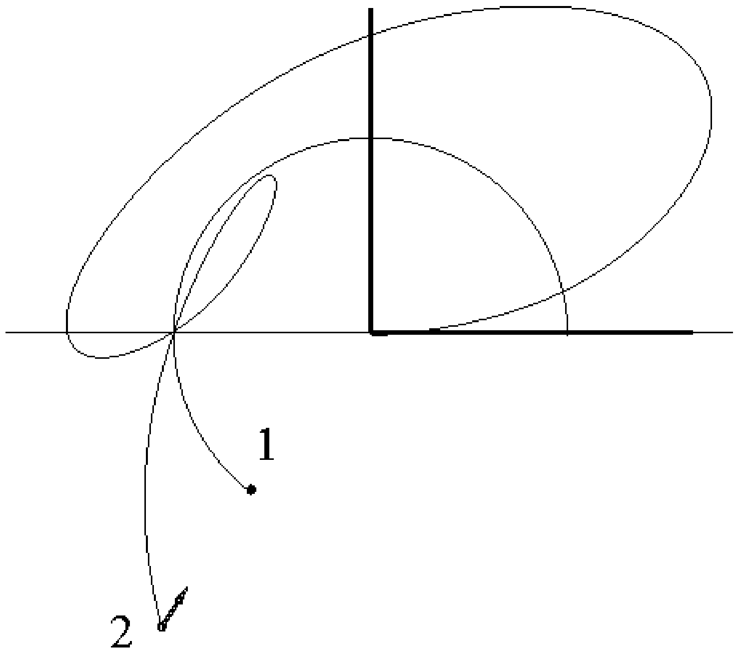


Figure 7.35. Pursuit game

7.14.1. Bodies

The model has two bodies represented by ellipsoids of radii **rad=0.05** and **rad2=0.05** correspondingly. Inertia parameters of the target (body 1) are not significant because of its kinematically defined motion (see below). The mass of the pursuer (body 2) is m2 (=3).

7.14.2. Joints

Motion of target is given with the generalized joint having three elementary transformations of type **tt**:

Elementary transformation	Axis	Vector	Functional expression
1	x	(1,0,0)	0
2	y	(0,1,0)	rcirc*cos(om*t)
3	z	(0,0,1)	rcirc*sin(om*t)

Here **rcirc=2** is the radius of the circle (target trajectory), and **om=1** is the angular frequency of the motion. Motion of pursuer is given with the six-degrees-of-freedom joint without rotational d.o.f. (they are turned off), thus the 2nd body has three translational d.o.f.

7.14.3. Forces

To simplify the model, the gravity force is turned off. To do this, set the gravity vector to (0,0,0) in **Object – Direction of gravity**. The strategy of the pursuer is defined by a bipolar force. It acts between the bodies **Pursuer** and **Target** and is of the type **Expression** having the constant value –**Force**. The negative sign means that the force decreases the distance between the bodies. The value of the **Force** is 50.

7.14.4. Simulation

Simulation results show (Figure 7.35) that such a simple pursuit strategy does not guarantee the target catching because of inertia forces: the pursuer all the time misses beside the target. The more effective strategy would be control by means of a predicted force (taking into account velocities of the bodies).

7.15. Bevel gearing

The model (Figure 7.36) consists of two bevel gears. The rotation axis of the large gear is fixed, that of another one (satellite) is moveable. The axes are perpendicular.

UM Model: LIBRARY\Gears.

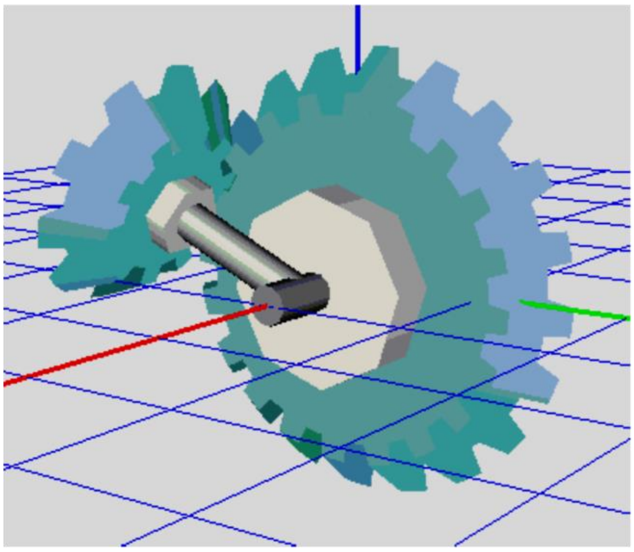


Figure 7.36. Bevel gearing

7.15.1. Gear images

To assign a gear image (body 1) create a new GO, and add a new GE, set the Parametric type and choose a standard element **Gear** in the pull-down box. Initialize the element parameters:

- rgear (=0.2) – radius;
- wgear (=0.05) – thickness;
- hgear (=0.025) – tooth height;
- zgear (=20) – number of teeth;
- tgear (=1) – gear taper.

For the second body make the same and set the parameters as follows:

rgear1=0.1; **wgear1**=0.05; **hgear1**=0.025; **zgear1**=10, **tgear**=1.

The system has the third body (Carrier), which connects the axes of gears. Its image be a cylinder with a length **rgear** positioned along the local Y axis of the body fixed SC.

Assign the material "steel" for all images, and turn on the autodetection of inertia parameters for all the bodies.

7.15.2. Joints

- Base0 – Gear, rotational joint, rotation axes for both bodies are X (1,0,0).
- Base0 – Carrier, rotational joint, rotation axes for both bodies are X (1,0,0).

- Carrier – Satellite, rotational joint, rotation axes: X (1,0,0) for Satellite, Y (0,1,0) for Carrier.

7.15.3. Forces

The model has a single force element, which simulates the gearing (**Special forces** menu item). Set its type to **gearing**.

Force element connects the Gear and the Satellite bodies.

Characteristic points are the centers of the mass (0,0,0) for both gears. The aces of rotation are the local X axes (1,0,0).

Set the rest gearing parameters:

- gear ratio to 2;
- clearance to 0;
- damping coefficient cdiss to 1000;
- tooth stiffness coefficient cstiff to 100000.

7.16. Deformable molecule

The model (Figure 7.37) contains a non-fixed body represented by a graphical image, which is changed in time.

UM Model: Molecule.

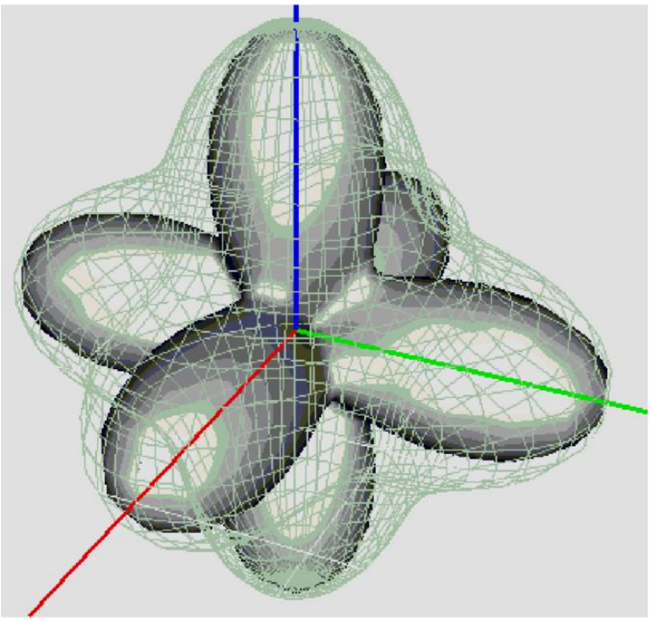


Figure 7.37. Molecule

7.16.1. Graphical image

UM graphical images (objects) are usually supposed to be rigid. However, if it is necessary (e.g. for demonstration), any GO can be being changed at simulation-time. For this purpose, its geometrical parameters should depend on some symbolic identifier, and then that identifier should be simulation-time changed by means of programming in the control file.

In the current model, **graphical object** contains a single graphical **element** of the type **parametrical** and the standard subtype **molecule**. Its Cartesian equations are

$$(x^2 + y^2 + z^2)^{n+1} = (x^n + y^n + z^n)^2$$

and depend on a parameter **mdeg** (exponent n) defining the molecule form.

Let **mdeg** be changed as

$$m_{deg} = \frac{n_{max} + n_{min}}{2} + \frac{n_{max} - n_{min}}{2} \sin \omega t,$$

that is from **nmin** (=1.2) to **nmax** (=5); let angular frequency ω be **om** (=5).

7.16.2. Body

Add a new body and link it with any joint without degrees-of-freedom, for example, with the 6-d.-o.-f. joint (turn off all the d.-o.-f.-s). Model is complete. Generate equations of motions.

7.16.3. Programming control file

Open the automatically generated control file (using the **Tools – Control file...** menu item) and modify its procedures as follows:

```
var
  IDind: integer;

procedure TimeFuncCalc( _t : real_; _x, _v : VectRPtr; _isubs : integer );
var
  _ : _moleculeVarPtr;
begin
  _ := _PzAll[SubIndx[_isubs]];
  _.mdeg := (_.nmax+_.nmin)/2 + (_.nmax-_.nmin)/2*sin(_.om*_t);
  SetIdentifierValue(IDind, 1, _.mdeg);
end;

procedure UserCalc( _x, _v, _a : VectRPtr; _isubs, _UMMessage : integer; var
WhatDo : integer );
var
  Key,i : integer;
begin
  Key := WhatDo;
  WhatDo := NOTHING;
  case _UMMessage of
    0 : begin
    end;
    FIRSTINIT_MESSAGE : begin
      GetElementIndexByName(eltIdentifier,  'mdeg', IDind, i);
    end;
    istep_end  : RefreshElement(eltGo,1,1);
    integr_end : RefreshElement(eltGo,1,1);
    xvastep_message : RefreshElement(eltGo,1,1);
    xvaend_message : RefreshElement(eltGo,1,1);
  end;
end;
```

Then compile the equations and run the simulation module.

7.17. Joint type conversion. Parameterization of axis inclination

Example: 1) illustration of joint type conversion (Chapter 3, Sect. Input of joints | Converting joint type);

2) parameterization of joint axis inclination.

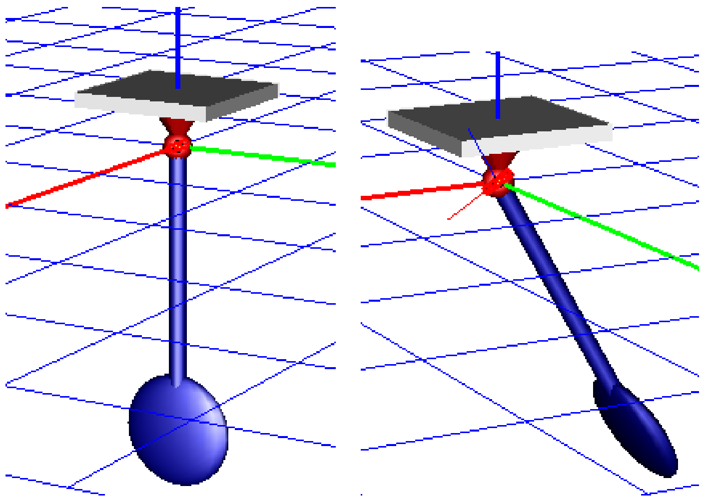


Figure 7.38. Pendulum with horizontal and inclined rotation axis

UM Model: LIBRARY\Pendulum; LIBRARY\Pendulum_inclined.

Read the *Library/Pendulum* model in the **UM Input** program. This is the model of a pendulum, which oscillates about the horizontal axis. Let us make changes to incline the rotation axis. The inclination angle should be parameterized, i.e. its value should be set using identifiers.

1. Save the model with a different name.

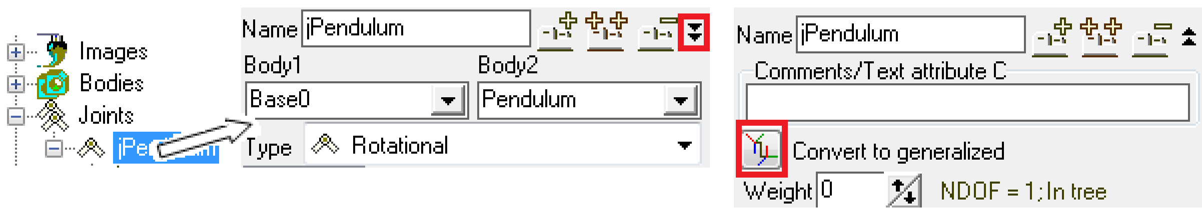
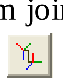


Figure 7.39. Conversion of rotational joint to generalized type

2. Open the jPendulum joint and convert it to the generalized type. First, click on the ▾ button and then on the  one, Figure 7.39.

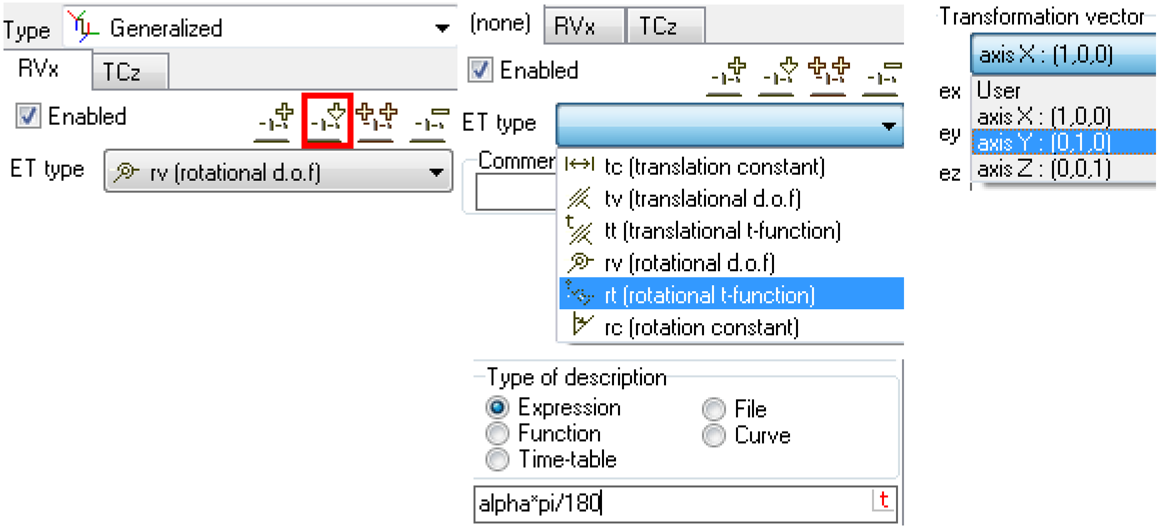


Figure 7.40. Adding ET parameterizing the inclination of the rotation axis

3. The rotational joint has been converted to the generalized one with two elementary transformations (ET). By active ET RVx (rotational degree of freedom relative to X axis) a new ET by the button  is added. Set its type *rt* (rotation as a time function). Select axis Y as the direction of the transformation vector. Finally, enter the expression *alpha***pi*/180 parameterizing the inclination of the rotation axis. The identifier *alpha* is the inclination angle in degrees.

The user can compare his model with the ready LIBRARY\Pendulum_inclined model, Figure 7.38.

7.18. Modeling of proportional friction with the use of block editor.

Example: leaf spring.

UM Model: Library/LeafSpring

From a mechanical point of view the car, passing uneven road surface, is an oscillating system, parts of which (wheels, suspension elements and body) oscillate.

A spring (from the French. ressort - elasticity, spring) is a type of a shock-absorbing device, the elastic suspension element of a transport machine, which transmits the load of the body on the undercarriages, wheels, caterpillars, etc., and which softens the shocks and blows while moving on uneven road.[1]

The model (Figure 7.41) is a body, united with the base by a translational joint-hinge. Leaf spring here simulates a spring which is loaded statically (with the force of gravity). The oscillation occurs due to the friction force caused between the spring leaves at their relative movement. In the input program value of the friction force is given by the identifier (Figure 7.42), and is initially zero.

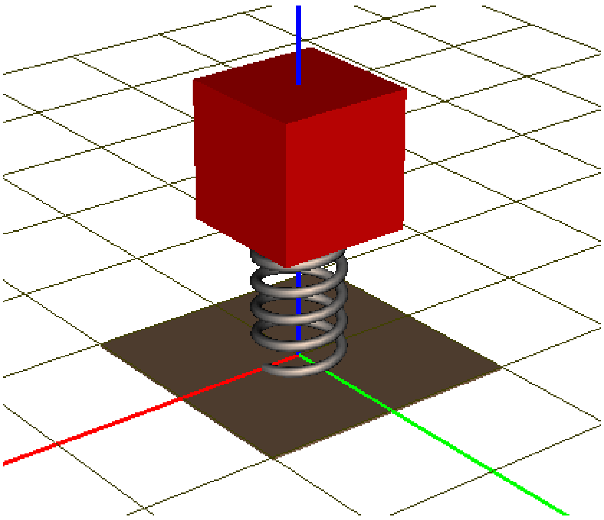


Figure 7.41 General view of the model

In the "Universal mechanism" software package leaf spring can only be described with a linear characteristic of the axial force (Figure 7.43). Proportional friction is implemented using the block editor (Figure 7.44).

[1] Large soviet encyclopedia

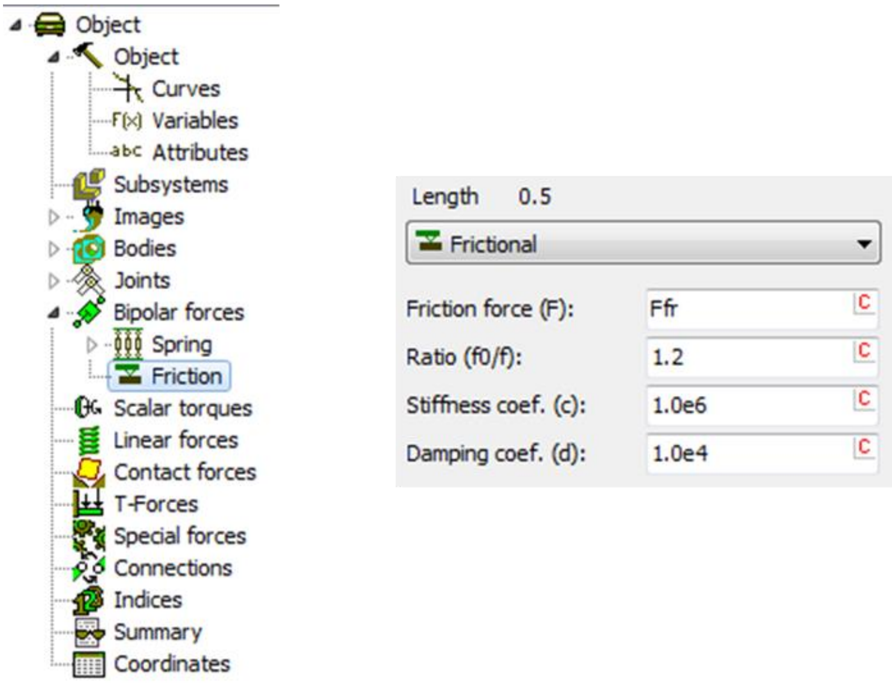
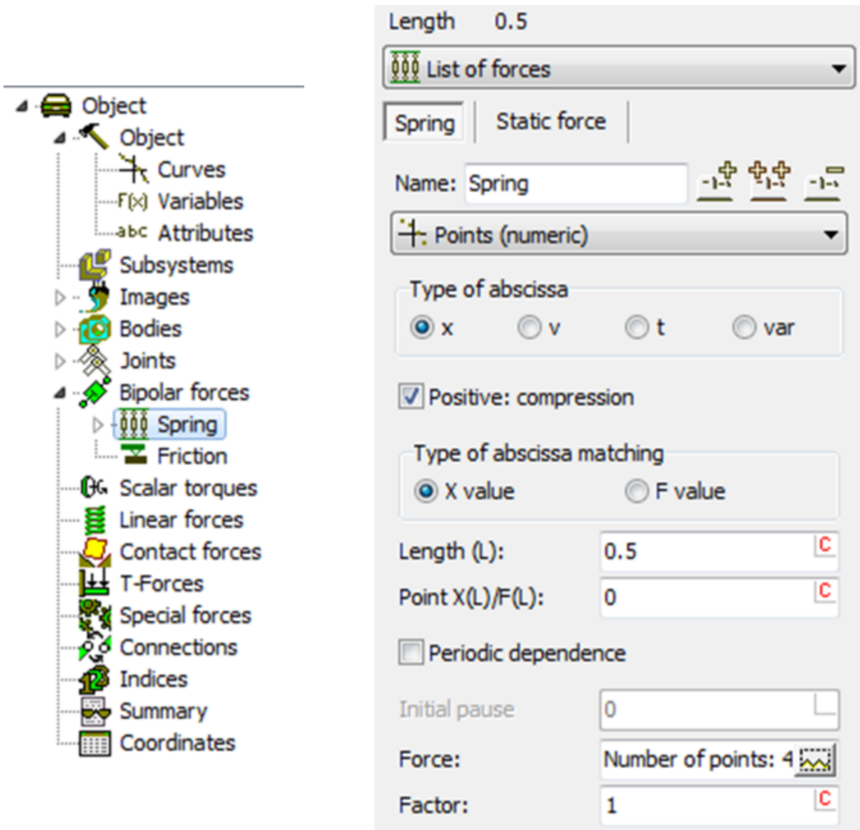


Figure 7.42 Setting frictional force



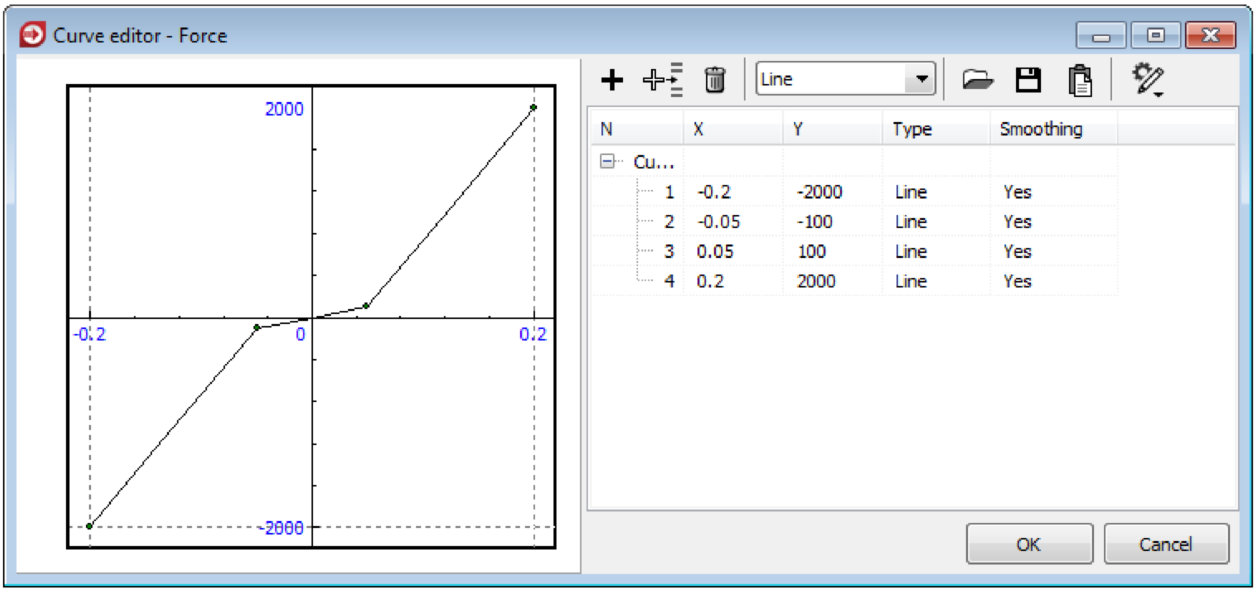


Figure 7.43 Axial force

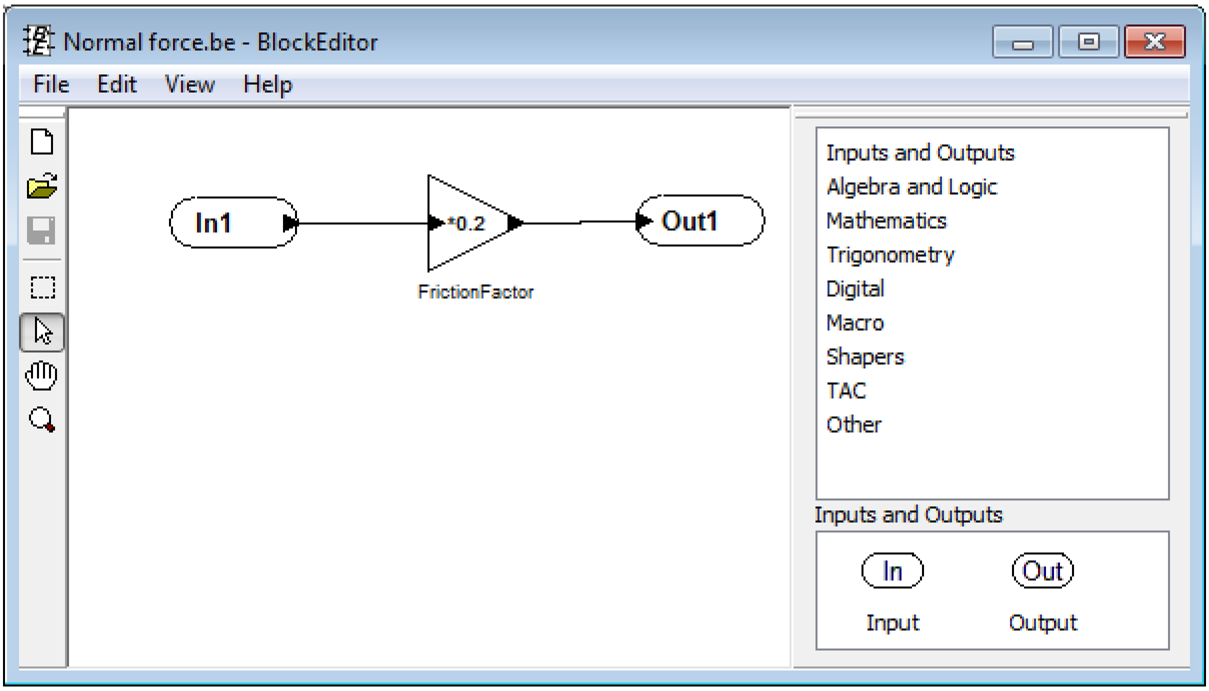


Figure 7.44 Block editor

In the simulation program using the Wizard of external libraries (Figure 7.45), the value of vertical force is on the input, inside the library it is multiplied by the coefficient of friction, and in the output we get the value of the friction force, which is assigned to the identifier.

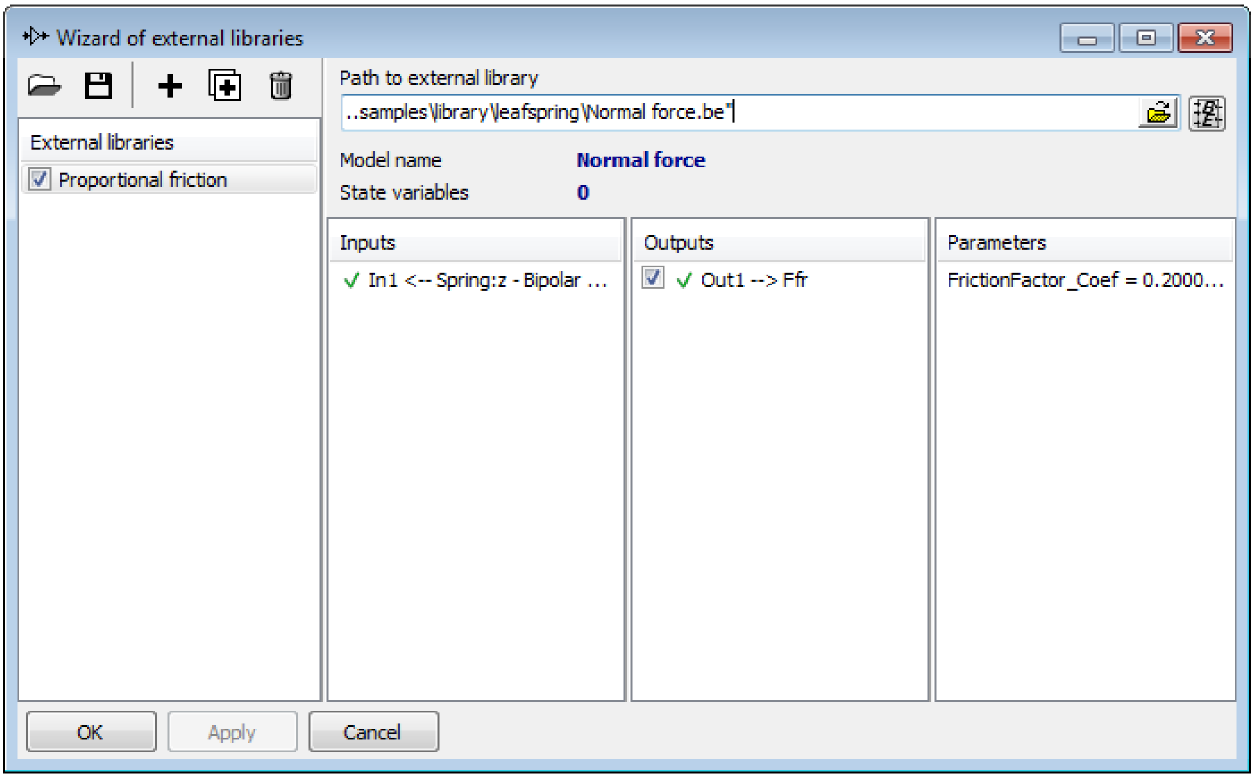


Figure 7.45 Wizard of external libraries

When you run the simulation, you can observe how the sliding friction force goes into the adhesion mode (Figure 7.46).

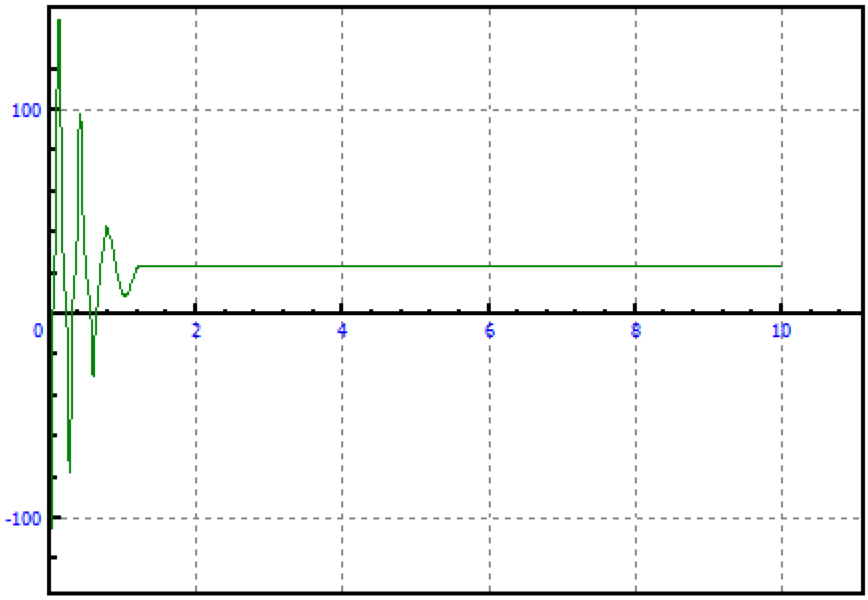


Figure 7.46 Friction force