Getting Started Using Universal Mechanism Software

Getting started

2019
Getting Started Using Universal Mechanism

This manual leads you through the basic possibilities of Universal Mechanism software and shows you how to create and simulate models of several simple mechanical systems. It assumes that you go through the manual step by step sequentially.

Simulation of such mechanical systems as cars and railway vehicles has certainly its own features but basic concepts using UM still the same. These concepts are shown in this manual.

Copyright and trademarks

This manual is prepared for informational use only, may be revised from time to time. No responsibility or liability for any errors that may appear in this document is supposed.

Copyright © 2019 Computational Mechanics Ltd. All rights reserved.

All trademarks are the property of their respective owners.

Contact information


Please, send your bug reports, questions and suggestions to um@universalmechanism.com.

Address:

Computational Mechanics Ltd.
Vostochnaya str. 2-14, Glinischevo, Bryansk region, 241525, Russia
Phone, fax: +7 4832 568637.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>MODEL OF PENDULUM</td>
<td>5</td>
</tr>
<tr>
<td>1.1.</td>
<td>WHAT WE WILL LEARN</td>
<td>5</td>
</tr>
<tr>
<td>1.2.</td>
<td>MODEL SCHEME</td>
<td>6</td>
</tr>
<tr>
<td>1.3.</td>
<td>CREATING THE MODEL</td>
<td>7</td>
</tr>
<tr>
<td>1.3.1.</td>
<td>Running UM Input and creating new model</td>
<td>7</td>
</tr>
<tr>
<td>1.3.2.</td>
<td>Familiarizing yourself with Universal Mechanism</td>
<td>7</td>
</tr>
<tr>
<td>1.3.3.</td>
<td>Creating graphical objects</td>
<td>8</td>
</tr>
<tr>
<td>1.3.3.1.</td>
<td>Scene image</td>
<td>8</td>
</tr>
<tr>
<td>1.3.3.2.</td>
<td>Image of pendulum</td>
<td>14</td>
</tr>
<tr>
<td>1.3.4.</td>
<td>Creating rigid bodies</td>
<td>15</td>
</tr>
<tr>
<td>1.3.5.</td>
<td>Creating joints</td>
<td>16</td>
</tr>
<tr>
<td>1.3.6.</td>
<td>Saving the model</td>
<td>17</td>
</tr>
<tr>
<td>1.3.7.</td>
<td>Run UM Simulation program</td>
<td>17</td>
</tr>
<tr>
<td>1.4.</td>
<td>SIMULATION OF MOTION</td>
<td>18</td>
</tr>
<tr>
<td>2.</td>
<td>FREE AND FORCED OSCILLATIONS</td>
<td>25</td>
</tr>
<tr>
<td>2.1.</td>
<td>WHAT WE WILL LEARN</td>
<td>25</td>
</tr>
<tr>
<td>2.2.</td>
<td>MODEL SCHEME</td>
<td>25</td>
</tr>
<tr>
<td>2.3.</td>
<td>CREATING THE MODEL</td>
<td>26</td>
</tr>
<tr>
<td>2.3.1.</td>
<td>Running UM Input and creating new model</td>
<td>26</td>
</tr>
<tr>
<td>2.3.2.</td>
<td>Creating graphical objects</td>
<td>27</td>
</tr>
<tr>
<td>2.3.3.</td>
<td>Creating rigid bodies</td>
<td>31</td>
</tr>
<tr>
<td>2.3.4.</td>
<td>Creating joints</td>
<td>33</td>
</tr>
<tr>
<td>2.3.5.</td>
<td>Creating force elements</td>
<td>36</td>
</tr>
<tr>
<td>2.3.6.</td>
<td>Visualization of spring and damper</td>
<td>37</td>
</tr>
<tr>
<td>2.3.7.</td>
<td>Additional parameters</td>
<td>39</td>
</tr>
<tr>
<td>2.3.8.</td>
<td>Preparation for simulation</td>
<td>41</td>
</tr>
<tr>
<td>2.4.</td>
<td>SIMULATION OF MOTION</td>
<td>42</td>
</tr>
<tr>
<td>2.4.1.</td>
<td>Free oscillations</td>
<td>42</td>
</tr>
<tr>
<td>2.4.2.</td>
<td>Statistical analysis</td>
<td>48</td>
</tr>
<tr>
<td>2.4.3.</td>
<td>Static and linear analysis</td>
<td>49</td>
</tr>
<tr>
<td>2.4.4.</td>
<td>Forced oscillations</td>
<td>51</td>
</tr>
<tr>
<td>3.</td>
<td>CANTILEVER BEAM</td>
<td>53</td>
</tr>
<tr>
<td>3.1.</td>
<td>WHAT WE WILL LEARN</td>
<td>53</td>
</tr>
<tr>
<td>3.2.</td>
<td>MODEL DESCRIPTION</td>
<td>53</td>
</tr>
<tr>
<td>3.3.</td>
<td>CREATING THE MODEL</td>
<td>54</td>
</tr>
<tr>
<td>3.3.1.</td>
<td>Creating graphical objects</td>
<td>54</td>
</tr>
<tr>
<td>3.3.2.</td>
<td>Creating bodies</td>
<td>55</td>
</tr>
<tr>
<td>3.3.3.</td>
<td>Creating joints</td>
<td>56</td>
</tr>
<tr>
<td>3.3.4.</td>
<td>Preparation for simulation</td>
<td>59</td>
</tr>
</tbody>
</table>
3.4. SIMULATION OF MOTION

3.4.1. Calculation of the equilibrium position
3.4.1.1. Calculation of equilibrium position by numerical solution of equilibrium equations
3.4.1.2. Calculation of the equilibrium position by integrating the equations of motion
3.4.2. Calculation of natural frequencies and modes of oscillations
3.4.3. Euler's critical load
3.4.4. Large deflections

4. SUBSEQUENT STUDYING OF UNIVERSAL MECHANISM

5. REFERENCES
1. Model of pendulum

1.1. What we will learn

This lesson shows you how to create new model, add rigid bodies and joints, generate and compile equations of motion, simulate dynamics of a model and obtain plots of various performances of the model. This lesson is devoted to general overview of the UM possibilities and workflow.

At the end of the lesson we will have the model of the pendulum (you can find the final model in the \{UM Data\}\SAMPLES\TUTORIAL\pendulum directory)\(^1\), which will include one rigid body – pendulum itself, one rotational joint and graphical object of the environment – support, see in Figure 1.1. After describing the model we will go through the all stages of the working with the model: synthesis and compiling of equations of motion, and then will come to the simulation of motion of the pendulum.

![Pendulum and Support](image)

Figure 1.1. Complete model

Then we will create the model of the multi-link pendulum by the development of the pendulum model and with the usage of the subsystem technique. We will learn how to create compound models, and features of working with mechanical systems with closed kinematical loops.

---

\(^1\) Pendulum model is also available at [http://www.universalmechanism.com/download/80/eng/pendulum.zip](http://www.universalmechanism.com/download/80/eng/pendulum.zip)
1.2. Model scheme

Before modeling the pendulum with the help of UM we recommend to draw its sketch like you can see in Figure 1.2.

As you can see, we drew a simple pendulum and chose two systems of coordinates (SC) – the base frame $O_XO_YO_Z$ (SC0) and the body-fixed frame (SC1). The SC0 origin is placed in the center of the joint, the second one (SC1) – at the mass center of the pendulum. The axes of SC1 are directed along the pendulum principle axes of inertia. The base frame exists in every object and, as a rule, is connected with the Earth. There is only rotational joint connecting the pendulum and the base frame (the wall which the pendulum is attached to).

![Figure 1.2. Pendulum scheme](image-url)
1.3. Creating the model

1.3.1. Running UM Input and creating new model

Running UM Input program

1. Click Start | Programs | Universal Mechanism 8.0 | UM Input.

Creating a new model

1. From the File menu point to New object.
The window of the constructor appears, see in Figure 1.3.

1.3.2. Familiarizing yourself with Universal Mechanism

Take a few minutes to familiarizing yourself with the Universal Mechanism constructor window, see in Figure 1.3.

![Constructor window](image)

**Tree of elements** of a model in the left top corner of the constructor window is used for getting access to elements of the model.

**Animation window** in the center shows the model or its elements. A frame is shown in the center of animation window. There is the following identification for axes: Red – X, Green – Y, Blue – Z (RGB). Point of view, zoom and other settings can be changed via toolbar buttons. Using the context menu you can set perspective parameters, supporting grid, etc.

**Inspector** at the right-hand side of the constructor is the main tool for the description of elements. It shows parameters of an active element. It contains full information about current element of the model.
1.3.3. Creating graphical objects

We recommend to start describing any mechanical system with creating a set of graphical objects (GO) of the elements of the model.

1.3.3.1. Scene image

Creating new graphical object – scene

Scene is a graphical object corresponding to fixed elements of the object. Describing the scene is optional. To create a scene you should make a usual graphical object and assign it to the scene image. As for our example it is an image of the fixed joint where the pendulum is attached to – support. In order to create the corresponding image you should do the following steps.

1. Point to Images element of Tree of elements.

Click button in the top of the Inspector to create new graphical object, see in Figure 1.4.

Note. You can add new element of any type in the same way.
Renaming the graphical object

As you create objects, UM automatically assigns names to them. Each name consists of a string containing the element type and a unique integer ID for that type. UM named the recently created graphical element GO1.

Point to the box with the name of the element and replace GO1 with Support, see in Figure 1.5.
Creating graphical elements

Every graphical object (GO) can include any number of various graphical elements (GE). So you are able to create quite complicated images. Let's create three elements – sphere, cone and box, which form the image of the support altogether.

Creating new graphical element: sphere

Click Add new graphical element button, see Figure 1.5. New tab GE1 appears, see in Figure 1.6.

1. Choose type for the new graphical element – Ellipsoid.
2. Point to the Parameters tab and set $a = b = c = 0.05$.
3. Point to the Color tab and set diffuse color to red.
Creating new graphical element: cone

1. Create new graphical element and set its type to **Cone**.

   **Note:** Do not add new graphical object instead new graphical element within graphical object. In this example we create the only graphical object – **Support**, which contains three graphical elements: sphere, cone and box.

2. Point to **Parameters** tab and set $R2 = 0.1; \ R1 = 0; \ h = 0.15$.
3. Set diffuse color to red.
Creating new graphical element: box

1. Create new graphical element and set its type to Box.
2. Point to Parameters tab and set \( A = 0.5; \ B = 0.5; \ C = 0.05. \)
3. Point to GE Position tab. Set Translation \( Z \) to 0.15, see in Figure 1.7.

![Figure 1.7. Graphical element position](image-url)
Assigning Support as scene image

1. Point to Object item of Tree of elements.
2. Select Support in the Scene image list, see in Figure 1.8.

Figure 1.8. Scene image choice
1.3.3.2. Image of pendulum

1. Return to the Images item in the Inspector.
2. Create new graphical object.
3. Rename new graphical object to Pendulum.

Note: Do not forget to press Enter after any modification of the text data in order to reflect this.

The pendulum image consists of two graphical elements: an ellipsoid and a cone.

4. Add new graphical element Ellipsoid and set its parameters $a = 0.05; b = 0.2; c = 0.2$. Set diffuse color to blue.
5. Add new graphical element Cone and set its parameters $R2 = 0.03; R1 = 0.03; h = 1$. Set diffuse color to blue.

Now image of the pendulum is ready.
1.3.4. Creating rigid bodies

The pendulum as a mechanical system consists of the only body.

1. Point to the Bodies item in Tree of elements.
2. Create new body by clicking the Add new element button, see Figure 1.9.

![Add new element](image)

Figure 1.9. Adding new body

3. Rename body to Pendulum.
4. Select Pendulum from the drop-down list Image.
5. Set Mass = 1 (kg), see in Figure 1.10.

![Pendulum creation](image)

Figure 1.10. Creating Pendulum
1.3.5. Creating joints

The rotational joint connects the Pendulum and the Base0. To create new joint do the following actions:

1. Point to Bodies | Pendulum.
2. Click the button Go to element and then select Create joint | Rotational to create a rotational joint, see Figure 1.11.

After that the rotational joint is created and named as jPendulum automatically. Joint points and joint vectors describe the position of the rotation axis relative to each of the bodies. Their coordinates must be given in the corresponding body-fixed systems of coordinates.

3. In the group Joint points | Pendulum set Z position to 1. So the pendulum will swing around its upper point, see in Figure 1.12.
1.3.6. Saving the model

Now your model is described completely. And it is high time to save it. Let the object name be **Pendulum**.

1. Select menu item **File | Save as…**
2. Set `{UM Data}\My models\Pend`, in the way how it is shown in the Figure 1.13.

![Image](image_url)

Figure 1.13. Data saving

1.3.7. Run UM Simulation program

The model is ready to be loaded in the **UM Simulation** program.

1. Select **Object | Simulation** menu item. UM Simulation program starts and opens the current model.
1.4. Simulation of motion

Now we are in the simulation program. We will open new animation window, deflect the pendulum from vertical position to 1 radian and run simulation of dynamics of pendulum.

Creating new animation window

From the Tools menu, select Animation window. New animation window appears. Familiarize yourself a bit with an animation window.

Rotating

Point the mouse cursor to the animation window so that cursor looks like the picture in the figure to the right. Press left mouse button and rotate the model in the animation window.

Shifting

Point the mouse cursor to the animation window so that it has Rotating shape, press Ctrl key and mouse cursor changes to Shifting mode. Press left mouse button and shift model in the animation window.

Zoom in/zoom out

Point the mouse cursor to the animation window and press Shift key and with the help of left mouse button zoom in/out the model. You can also use mouse scroll wheel.

After some practice you can get something like shown in the Figure 1.14.

Figure 1.14. Pendulum model in the animation window
Start simulation

1. From the Analysis menu, select Simulation. Window of the **Object simulation inspector** appears, see in Figure 1.15.

![Object simulation inspector window](image)

Figure 1.15. **Object simulation inspector** window
Initial conditions

You should deflect the pendulum a bit in order to obtain its motion. There exists a special tool for this purpose: a wizard of the initial conditions.

1. Select the **Initial conditions | Coordinates** tab.
   You can see a complete list of the object coordinates. In our case there is only one coordinate in **jPendulum** joint.

2. Set **Coordinate** to **1**. Press **Enter** key, see Figure 1.16.
   Your pendulum has deflected.

**Note:** Universal Mechanism uses *System International (SI)*. Angular values have dimensions of *radian*.

![Figure 1.16. Pendulum movement initial conditions](image-url)
Simulation

Now your model is ready for simulation. Simply start simulation process for the 10 seconds.

1. Click the Integration button in Object simulation inspector. Process parameters window will appear at the lower right corner of the screen, see in Figure 1.17.

![Figure 1.17. Integration process parameters](image)

At the end of the simulation the Pause window appears, see in Figure 1.18. You can increase the simulation time, change the numerical method etc.

![Figure 1.18. Pause window](image)

2. Press the Interrupt button. Object simulation inspector appears.
Drawing plots

During the simulation you can see plots of various variables. Such as velocities, accelerations, forces and so on. We will open new graphical window, create new variable to plot – Y coordinate of the center of mass of the pendulum and draw its plot.

Well, let us create new graphical window.

1. From the Tools menu, select **Graphical window**.
   Open **Wizard of variables**.

2. From the Tools menu, select **Wizard of variables**.

The **Wizard of variables** is a special tool for creating variables, which can be drawn in graphical windows or animated in animation windows (in cases of vectors or trajectories), see in Figure 1.19.

![Wizard of variables window](image)

**Figure 1.19. Wizard of variables** window
Let us draw a plot of Y coordinate of the mass center of the pendulum.

3. Select **Linear variables** (coordinates, velocities, accelerations).
4. Select **Y** in the **Component** group.
5. Then move the variable to the container with the help of the button

   New variable $r\cdot y(Pendulum)$ appears in the container of variables.
6. Select the variable in the **Wizard of variables** and drag it to the graphical window.
7. Select the **Object simulation inspector** and click the **Integration** button.
   You can see the plot of your variable in the graphical window, see in Figure 1.20.

![Figure 1.20. Creating the plot of the variable in the graphical window](image)

**Animation of vectors and trajectories**

During the simulation you can animate various vector variables in an animation window. Let us animate the vector of the mass center velocity. Firstly, we need to create such variable in the **Wizard of variables**.

1. Select the **Wizard of variables** and there select the **Linear variables** tab.
2. Select **Velocity** in the **Type** group.
3. Select **V** (vector) in the **Component** group.
4. Add this variable to the container clicking the button.
5. Drag new variable to the animation window.
   A list of animated vectors is hidden by default. You can make it visible and change its position with the help of the **Position of list of vectors** command of the pop up menu of the animation window.
6. Select animation window, click right mouse button and select **Position of the list of vectors**
   | **Bottom**
   To draw a trajectory of the pendulum create a new variable with the help of the master.
7. Repeat all steps we made for the velocity, but **Type** of the variable set to **Coordinate**. Drag this variable to the animation window.

8. Double click on the velocity item in **List of vectors** and select red color for the vector of velocity and then double click trajectory item and select blue color for it.

9. Click the **Integration** button in **Object simulation inspector**.

   Now you can see the vector of the velocity and trajectory of the center of the mass of the pendulum, see in Figure 1.21. You should use the **Scale of vectors** command of a pop up menu to specify its scale.

   Double click on an element of the list of vectors or on a vector/trajectory image to change the color of the vector and trajectory (in addition for the trajectory – to change the number of points on the curve).

---

**Figure 1.21.** Velocity and trajectory vector of the pendulum animation
2. Free and forced oscillations

2.1. What we will learn

In this lesson we will learn how to add forces, preset movement of a body as a time function and use parameterization of a model. We will use Linear analysis for obtaining the equilibrium position of a system, natural frequencies and forms. As well as we will analyze the spectrum of output data using the Statistics tool.

2.2. Model scheme

The example of simulation of free and forced damped oscillations is considered. In this lesson we will create the model shown in the Figure 2.1. Model consists of two rigid bodies Top and Brick, two translational joints, a linear spring and a damper. We will set vertical coordinate of the upper body as a sinusoid function.

You can find the final model in the [UM Data]\SAMPLES\TUTORIAL\oscillator directory or download it using the following link:


![Figure 2.1. Model scheme](image-url)
2.3. Creating the model

2.3.1. Running UM Input and creating new model

Running UM Input program

1. Click Start | Programs | Universal Mechanism 8.0 | UM Input.

Creating a new model

1. From the File menu point to New object. The window of the constructor appears.
2.3.2. Creating graphical objects

**Top**

We will create a thin rectangular plate as a graphical object for the *Top* body.

1. Create new graphical object.
2. Set its name to *Top*.
3. Add new graphical element – *Box*.
4. Set parameters and GE position for the *Box* as it is shown below.
5. Select the *Color* tab and choose *blue* for diffuse and specular colors, see in Figure 2.2.

![Creating the Top graphical object](image)

Figure 2.2. Creating the *Top* graphical object
**Brick**

Let us describe the brick as a cube of **0.2 m** side length.

1. Create new graphical object.
2. Set its name to **Brick**.
3. Add new graphical element **Box** into this graphical object.
4. Set its parameters and GE position, see in Figure 2.3.
5. Select the **Color** tab and set **red** for **diffuse** and **specular** colors.

![Creating the Brick graphical object](image)

Figure 2.3. Creating the **Brick** graphical object
Now we will create the graphical object for the spring.

1. Create new graphical object.
2. Set its name to Spring.
3. Add new graphical element Spring.
4. Set Spring parameters as it is shown in Figure 2.4.
5. Select the Color tab and set yellow for diffuse color and red for specular color.

Figure 2.4. Creating the Spring graphical object
Damper

Now we come to the last graphical object in this model – damper, see in Figure 2.5.
1. Create new graphical object.
2. Set its name to Damper.
3. Add new graphical element for the Damper – Cone.
4. Set parameters as follows:
   \[
   \begin{align*}
   & R2 = 0.02; \\
   & R1 = 0.02; \\
   & h = 1.
   \end{align*}
   \]
5. Select the Color tab and choose blue for diffuse and specular colors.
6. Add one more Cone with the following parameters:
   \[
   \begin{align*}
   & R2 = 0.04; \\
   & R1 = 0.04; \\
   & h = 0.5.
   \end{align*}
   \]
   In the GE Position tab set Translation \( Z \) to 0.25. Select red diffuse and specular colors.

Figure 2.5. Damper graphical object
2.3.3. Creating rigid bodies

**Top**

Create new rigid body **Top**.
1. Add new rigid body.
2. Set its name to **Top**.
3. In the **Image** list select **Top**, see in Figure 2.6.
4. Leave **Mass** and **Inertia tensor** empty.

![Image](Image.png)

Figure 2.6. Creating the **Top** rigid body
Brick

Now we will create one more rigid body Brick. Its mass we will express via parameter (identifier) \( m \). Such a parameterization gives us a possibility to change its mass easily and quickly obtain results for various values of the mass of the brick without regeneration equations of motion. Otherwise we would have to generate equations every time we want to change its mass.

1. Add new rigid body.
2. Rename it to Brick.
3. In the Image list select Brick.
4. Set Mass to \( m \) and press Enter. New Initialization of values window appears.
5. Set Value to 10. Press Enter.
6. This new parameter appears in the parameter list in the bottom left corner of the constructor window, see in Figure 2.7.

![Figure 2.7. Creating the Brick rigid body](image_url)
2.3.4. Creating joints

Joint for the top

The **Top** body moves along the vertical direction according $A \cdot \sin(\omega \cdot t)$ function. Now we will describe the translational joint between the base and the top and set the coordinate in this joint as a time function.

1. Select the **Top** body in the tree of elements.
2. Click the button. Select **Create joint** and in the drop-down list select **Translational**, see the Figure 2.8, left. The new joint of this type is created. Now you can see parameters of the joint.
3. Select the **Geometry** tab and set joint parameters as it is shown in the Figure 2.8, right.

![Figure 2.8. Creating translational joint](image-url)
4. Point to the **Description** tab.

5. Turn on the **Prescribed function of time** check box.

6. Set **Type of description** to **Expression**, and then input \(a\sin(\omega t)\), see Figure 2.9, and press **Enter**.

![Figure 2.9. Prescribed function of time](image)

7. In the **Initialization of values** window set \(a = 0.05\) (m) and \(\omega = 10\) (rad/s), see in Figure 2.10.

![Figure 2.10. Initialization of values window](image)
Joint for the brick

1. Select the **Brick** body in the tree of elements.
2. Click the button.
3. In the drop-down list select **Translational** again.
4. Select the **Top** as the first body instead **Base0**, see in Figure 2.11.
5. Set the rest parameters of the joint as it is shown in Figure 2.11.

Figure 2.11. Joint for the **Brick** parameters
2.3.5. Creating force elements

Now we will describe elastic and damping force elements between the top and the brick. Let us use \( c \) parameter for the stiffness coefficient of the spring and \( \mu \) parameters for the damping coefficient of the damper. Length of the unloaded spring let us denote as \( \text{Length} \).

1. Select the \textit{jBrick} joint.
2. Select the \textit{Joint force} tab.
3. In the \textit{Joint force} list select \textit{Linear}.
4. In the \( c \) box input \( c \) (stiffness coefficient), in the \( x_0 \) box input \( \text{Length} \) and set \( d \) to \( \mu \), see Figure 2.12. Press \textit{Enter}. Set values of parameters as follows: \( c = 250 \), \( \text{Length} = 0.4 \), \( \mu = 5 \).

![Figure 2.12. Elastic and damping joint forces](image-url)
2.3.6. Visualization of spring and damper

After all we have completely described object from the mechanical point of view. We described all elements we need: rigid bodies, joints and force elements. However our model now looks not so good – spring and damper introduced as joint forces that cannot be visualized, see Figure 2.13, left. In order to visualize spring and damper we will create two bipolar forces in the model. Their values we set to zero. That is why these bipolar forces will not influence on the dynamics of the model, but give us a possibility to show the spring and the damper, see Figure 2.13, right.

![Visualization of forces](image)

Figure 2.13. Visualization of forces

**Note.** There are several possible ways to describe elastic and damping forces in our model. We used the way to describe them as joint forces, but it is not the only right way. We could introduce them as bipolar as well. And in this latter case we would visualize them and introduce forces at once without intricate describing additional fake bipolar forces. But such a way leads to the following problem. Our ideal case that we consider here allows to model the situation when the length of the spring and damper equal to zero. Imagine that the brick has so large amplitude that the attachment points of the spring and damper will be on the same level. In such a case we have degeneration of bipolar forces that act along the line between the attachment points. When we have zero length we could not find the direction of the bipolar forces. Joint forces have no such a degeneration, because they direction always coincide with the axis of the joint. That is why we used very joint forces here.
1. So, select **Bipolar forces** in the tree of elements.
2. Add two bipolar force elements. Set their parameters as it is shown in the Figure 2.14.

![Figure 2.14. Fictitious bipolar forces](image)

Figure 2.14. Fictitious bipolar forces
2.3.7. Additional parameters

Using UM you can express one parameter via others. Here we will add two new parameters in the model – accurate analytical values of the natural frequency and the critical damping coefficient. Our model is very simple that is why we can obtain analytical solutions easily.

Natural frequency can be obtained according to the following formula:

\[ k = \frac{c}{\sqrt{m}}, \]

where \( k \) is natural frequency, rad/s;
\( c \) is stiffness coefficient, N/m;
\( m \) is mass of the body, kg.

Critical damping coefficient can be found as:

\[ \mu^* = 2\sqrt{cm}, \]

where \( \mu^* \) is critical damping coefficient, Ns/m.

3. Well, add new identifiers (parameters) to our model. Click the button in the list parameters or select the New identifier menu command from the context menu, see in Figure 2.15.

4. Fill out the Add identifier form as it is shown in the Figure 2.16.
5. Add one more identifier: \( \text{mu}\_\text{star} = 2\sqrt{c\_\text{m}} \). It is a critical damping coefficient, see in Figure 2.17.
2.3.8. Preparation for simulation

1. Save the model as Oscillator (use menu command File | Save as…).
   Now we will come to the simulation program.

2. From the Object menu select Simulation… or simply press Ctrl+M.
   The simulation program starts and opens the current model.
2.4. Simulation of motion

Let us consider some particular cases of oscillations: free damped oscillations and forced oscillations without damping.

2.4.1. Free oscillations

Free damped oscillations

1. Open new animation window ([Tools] | Animation window…).
2. Open new graphical window, where we will plot time history of the vertical position of the brick.
3. Open Wizard of variables ([Tools] | Wizard of variables…).
4. Select the Linear variables tab, select Brick in the list of bodies, set Type to Coordinate, set Component to Z. Click the button to create new variable. The variable appears in the container of variables. Drag the variable to the graphical window. Close the Wizard of variables.
5. From menu Analysis select Simulation…. The Object simulation inspector appears.

Figure 2.18. Desktop of the simulation program
6. Arrange windows on the desktop as you prefer, for example, as it is shown in the Figure 2.18.

7. Select **Object simulation inspector** and click the **Identifiers** tab.

8. Set \( a \) to 0 and press **Enter**, see in Figure 2.19. So we set zero amplitude of the oscillations of the **Top** body, in other words we fix the **Top** body in order to analyze free oscillations.

![Object simulation inspector](image)

**Figure 2.19. Parameters of the model**
9. Select the **Initial conditions** tab. In **Coordinate | 1.1** input **0.1** see in Figure 2.20. We need to shift the brick a bit because its position at zero coordinate is quite near to its equilibrium position that gives us small amplitude of oscillations if we do not shift the body.

![Initial conditions coordinate tab](image)

**Figure 2.20. Initial conditions coordinate tab**

10. Select the **Solver** tab. Set **Simulation time** to **25** (seconds).

11. Run simulation clicking the **Integration** button.

Process of the numerical simulation lasts for 25 seconds. You can see oscillations of the brick in the animation window and time history of the vertical position of the brick.
12. Click the **100%** button in the drop-down tool panel in the top or click the **Show all** menu command in the context menu, see in the Figure 2.21. Plot now fits the window.

![Graphical window after the first experiment](image)

Figure 2.21. Graphical window after the first experiment
Free oscillations without damping

Now we will turn off damping and compare plots for damped and free oscillations. Using zero damping coefficient gives us free oscillations.

1. Select the graphical window. Point to the \texttt{r:z(\text{Brick})} variable in the list of variables. Open context menu. Select the \textbf{Copy as static variables} menu command. The second variable appears.

2. Select the \textbf{Pause} inspector and click the \textbf{Interrupt} button. \textbf{Object simulation inspector} appears.

\begin{tcolorbox}
\textbf{Note.} The \texttt{r:z(\text{Brick})} variable, which we dragged from \textbf{Wizard of variables}, will be recalculated for every numerical experiment. It is so called dynamic variable. In order to compare plots for different experiments we need to copy dynamic variables as static ones. Static variables are not changed from one experiment to another.
\end{tcolorbox}

3. Select \textbf{Object simulation inspector} and point to the \textbf{Identifiers} tab.

4. Set \texttt{mu = 0} and press \textbf{Enter}. So we have just turned off damping.

5. Click the \textbf{Integration} button.

It will take you some seconds to finish the simulation. In the Figure 2.22 you can see the graphical window after two numerical experiments.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{plot.png}
\caption{Graphical windows after the first experiment}
\end{figure}
Free oscillation: critical damping

As we showed above critical damping coefficient is \( \mu = 100 \) Ns/m. Let us analyze the motion of the Brick in such a case.

1. Point to the graphical windows. Select the first variable \( r:z(\text{Brick}) \) and copy it as a static one again (use the Copy as static variable item from the context menu).
2. Select the Pause inspector and click the Interrupt button. Object simulation inspector appears.
3. Select the Identifiers tab and set \( \mu = 100 \).
4. Click Integration.
   Now you can see that the motion of the brick is non-periodic, see in the Figure 2.23.

![Figure 2.23. Graphical window after three numerical experiments](image)

5. Make numerical experiments for other values of the damping coefficient. Do not forget to copy variables as static ones.
6. If you changed the value of the damping coefficient, set it again to \( \mu = 100 \) Ns/m.
2.4.2. Statistical analysis

Now we will come through some additional tools for analysis of results of the simulation.
1. From the **Tools** menu select **Statistics**…. New **Statistics** window appears.
2. Drag the variable, which corresponds to free oscillations, from the graphical window to the **Statistics** window.
3. Select the **Statistics** window and point to **Power spectral density**.
   The characteristic shape of the power spectral density shows the process has the only frequency, which corresponds to natural frequency. We have the accurate analytical solution – 5 rad/s. Not let us obtain this frequency numerically from the plot of the power spectral density, see the Figure 2.24. It is approximately **0.78 Hz**, see abscissa in the left bottom corner, 0.78 Hz gives us $0.78 \cdot 2\pi = 4.9$ rad/s. You can see that numerically obtained values are quite close to analytical one.

**Note.** To pick the frequency more precisely use changing scale of the window as it is shown in the Figure 2.24.

![Power spectral density of the free oscillations](image)
2.4.3. Static and linear analysis

Let us consider an example of using the Static and linear analysis. With the help of this tool we will find the equilibrium position of the system, its natural frequencies and modes, define how big the actual damping ratio is.

Well, at first you need to close the Pause and Object simulation inspector windows.

1. Select the Pause windows and click Interrupt.
2. Select the Object simulation inspector and click Close.

Open Linear analysis window

3. From the Analysis menu item select the Static and linear analysis… command. The Static and linear analysis window appears.

Equilibrium position

4. Select the Equilibrium tab. Click the button on the tool panel. You can see the message “Equilibrium position is successfully computed!”. The model in the animation window is now in its equilibrium position.

Note. Obtained coordinates, which correspond to the equilibrium position, can be saved to a file. To do it use the button in the Initial conditions tab. This file with initial conditions can be loaded later in the Object simulation inspector in order to start the simulation from the equilibrium position if necessary.

Natural frequencies and forms

5. Select the Frequencies/Eigenvalues tab and click the button to compute frequencies. The list of the natural frequencies is located in the left part of the window. As you can see our system has only one frequency 0.795775 Hz, what corresponds exactly to 5 rad/s.
6. Click the button to start animation of the natural modes. Adjust appropriate Amplitude and Rate. Click the button or Esc key to finish the animation.

Stability of equilibrium

Let us find the roots (eigenvalues) of the linearized system. It gives us the information about stability of the model.

7. Check the Eigenvalues option and click the button to compute the eigenvalues.
8. Eigenvalues are shown in the right table. For each of the pair of complex conjugated roots, only one value with positive imaginary part is shown in the table. Select the \textbf{Re/Im} mode of eigenvalues (real and imaginary parts). As you can see, the real part of the root is negative. Thus, the equilibrium position is asymptotically stable.

\textbf{Damping ratio}

9. Select the \textbf{Frequency/Damping ratio} mode of roots. We have the damping ratio value $\text{Beta} = 100 \%$, which corresponds to critical damping. The frequency should be theoretically equal to zero, but in reality it is very small due to approximated computations.

\textbf{Note.} Damping ratio says us whether all forms are damped properly and whether some change of damping coefficients or damper locations is necessary.

10. You can change the value of the \textbf{mu} identifier in the \textbf{Identifiers} tab and see what will happen with damping ratio and frequency.

11. Close the \textbf{Static and linear analysis} window.
2.4.4. Forced oscillations

Let us consider simulation of forced oscillations without damping.
1. Delete all variables from the graphical window except the first (dynamic) one.
2. From the Analysis menu select Simulation…
3. Select the Identifiers tab. Set the following values: $a = 0.05$, $\omega = 8$, $\mu = 0$.
4. Run integration. Now you can see that the body Top also moves. The time history of the vertical position of the Brick is given in the Figure 2.25.

![Figure 2.25. Forced oscillation (omega = 8 rad/s)](image-url)
Resonance

In conclusion we consider the resonance case, when the excitation frequency is equal to the natural frequency of the system.

1. In the Pause window click the Interrupt button.
2. In the Object simulation inspector set $\omega = 5$.
3. Run integration. As we expected in the resonance case the amplitude of the oscillations increases in the long run, see in the Figure 2.26.

![Figure 2.26. Forced oscillations: resonance case](image)
3. Cantilever beam

3.1. What we will learn

In this lesson we will consider the cantilever beam, for which we will find the equilibrium position, eigenfrequencies and mode shapes, Euler's critical load and a big deflection case.

3.2. Model description

For beam modeling we will use the method of separate bodies [1]. The beam is considered as a chain of rigid bodies (rods) interconnected with rotational joints. Rods at both ends have a length of $l/2$, and the length of internal rods is $l$, where $l = L/(N - 1)$, $L$ is the length of the beam, $N$ is the number of rods. The leftmost rod is rigidly connected to the support. Each pair of rods is connected by a rotational joint containing a linear elastic-dissipative force element with a stiffness coefficient $c$ and a damping coefficient $\alpha$, Figure 3.1.

![Figure 3.1. Elastic beam model as a chain of absolutely rigid bodies](image)

As an example let us consider the steel beam ($E = 2,1 \cdot 10^{11}$ Pa, $\rho = 7850$ kg/m$^3$) with round cross-section and length $L = 3$ m. The beam in the model is divided in 11 rigid rods. You can find the ready to use model in the following folder

{UM Data}\samples\tutorial\cantilever_beam

or download it using the following link:

3.3. Creating the model

Run UM Input program and create a new object and add the following model parameters as it is shown in the Table 3.1: BeamLength, YoungsModulus, l, d, J, cax, dax.

<table>
<thead>
<tr>
<th>Name</th>
<th>Expression</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BeamLength</td>
<td>3</td>
<td>Beam length</td>
</tr>
<tr>
<td>YoungsModulus</td>
<td>2.1000000E+11</td>
<td>Young's modulus</td>
</tr>
<tr>
<td>l</td>
<td>BeamLength/10</td>
<td>Length of rigid rod</td>
</tr>
<tr>
<td>d</td>
<td>2*l/75</td>
<td>Cross-section diameter</td>
</tr>
<tr>
<td>J</td>
<td>pi<em>d</em>d*d/d/64</td>
<td>Moment of inertia of cross-section</td>
</tr>
<tr>
<td>cax</td>
<td>YoungsModulus*J/l</td>
<td>Stiffness coefficient</td>
</tr>
<tr>
<td>dax</td>
<td>5</td>
<td>Damping coefficient</td>
</tr>
</tbody>
</table>

Table 3.1.

Beam model parameters

3.3.1. Creating graphical objects

The model will contain 4 graphical objects: one for the support and three for the different rods of the beam.

Support

Let us create a graphic image of the support.
1. Create a new graphic object like Box.
2. Use Support as a name for it and set the following parameters:
   
   \[
   \begin{align*}
   A &= 0.5; \\
   B &= 0.05; \\
   C &= 0.5. 
   \end{align*}
   
3. Go to the GE position tab. Set Translation | y to 0.025 and Translation | z to -0.25.
4. Click the Colors tab. Set for graphic element brown diffuse color.

Rods

Create graphic images of rods.
1. Create a new graphic object of Cone type.
2. Assign a name Rod1 for it and set the following parameters:
   
   \[
   \begin{align*}
   R2 &= d/2; \\
   R1 &= d/2; \\
   h &= l/2. 
   \end{align*}
   
3. Go to GE position tab. In Translation | y field type -l/4. In Rotation group choose X axis and set value -90, Figure 3.2.
5. Go to tab **Colors**. Set diffuse color for graphic element **blue**.
6. Create one more graphic object with graphic element of **Cone** type. Name it **Rod2** and set the following parameters:

   \[
   \begin{align*}
   R2 &= d/2; \\
   R1 &= d/2; \\
   h &= l.
   \end{align*}
   \]

7. Go to **GE position** tab. In **Translation** | y field type \(-l/2\). In **Rotation** group choose \(X\) axis and set value \(-90\).
8. Go to **Material** tab. In **Density** field set value **7850**.
9. Go to **Colors** tab. Set **yellow** diffuse color for the graphic element.
10. Make a copy of graphic object **Rod2**. Name it **Rod3** and set **blue** diffuse color.

![Image of graphic element settings](image)

**Figure 3.2.** Setting the position and orientation of the Rod1 graphic element

### 3.3.2. Creating bodies

**Support**

Create a new rigid body for support.

1. Add new rigid body.
2. Type **Support** in the **Name** field.
3. Select **Support** graphical object in the drop-down **Image** list.
4. Turn on **Compute automatically** inertia parameters using the image of the body mode, Figure 3.3.
3.3.3. Creating joints

Joint for support

The support is a fixed body, i.e. has no degrees of freedom. Let us introduce a joint, rigidly connecting the base and support.

1. In the tree of elements select **Bodies | Support**.
2. Click **Go to element** button and then select **Create joint**.
3. In the appeared list of joint types select 6 **d.o.f.** After that, a new joint of the specified type will be created, which has a support as the second body. By default this joint introduces six degrees of freedom for the body: three translational and three rotational degrees of freedom.
4. Assign **Base0** as the first body.
5. Turn off all joint degrees of freedom, thus connect rigidly the support and base, Figure 3.4.

![Joint for body Support](image)

**Figure 3.4. Joint for body Support**

**Joint between the first rod and support**

Create a joint that will satisfy the conditions of rigid clamping of the beam, i.e. prohibit any movement of its left end.
1. In the tree of elements choose **Bodies | Body1**.
2. Push **Go to element** and create joint of type **6 d.o.f.**
3. Name it **jSupportBody1**.
4. Assign **Support** as the first body.
5. Go to tab **Geometry | Body 2**. In the **Translation** | y field put $-l/4$.
6. Turn off all joint degrees of freedom, thus rigidly linking the rod and support.

**Joints between rods**

Now we will introduce rotational joints with elastic-dissipative forces between the rods.
1. In the tree of elements select **Bodies | Body2**.
2. Click the **Go to element** button create a joint of **Rotational** type.
3. Set **Name** to **jBody1Body2**.
4. Assign **Body1** as the first body.
5. On the **Geometry** tab for the first body (**Body1**) in the group **Joint points** input coordinates of its right end and for the second body (**Body2**) input coordinates of the left end, see Figure 3.5.
6. Go to the **Joint force** tab and select the **Linear** force type. Set **Stiffness coef.** (c) to cax, and set **Damping coef.** (d) to dax, Figure 3.6.

7. By analogy, create joints for bodies **Body3-Body11**.

![Figure 3.5. Rotational joint between Body1 and Body2](image1.png)

![Figure 3.6. Setting the elastic-dissipative force in the joint](image2.png)
3.3.4. Preparation for simulation

Check the correctness and completeness of the model description. In the tree of elements, select **Summary** (Ctrl+Alt+P). If the model is described correctly, the inspector will display a message that there are no errors, Figure 3.7. Modeling is not possible if there are errors. By clicking the left mouse button on the error or warning message line, you can go to the description of the corresponding element.

Save model under the name *cantilever_beam*. Go to **UM Simulation** program. To do this choose **Object | Simulation…** (Ctrl+M) in the main menu. **UM Simulation** with the beam model will run automatically.

![Figure 3.7. Analysis of correctness and completeness of the model description](image-url)
3.4. Simulation of motion

Let us consider the following problems: the calculation of the equilibrium position, the calculation of eigenfrequencies and mode shapes and the Euler's critical force.

3.4.1. Calculation of the equilibrium position

Generally, any model, created in the UM software, is not initially in the equilibrium position. To avoid at the start of modeling intensive transient processes, it is necessary to find the equilibrium position. As a rule, this is the first step to start working with a new model. The equilibrium position can be found using two ways:

- numerical solution of equilibrium equations;
- numerical integration of the equations of motion to get as close as possible to the equilibrium position.

The first method is usually faster and gives more accurate result. In addition, by numerically integrating the equations of motion, you can approach only the asymptotically stable equilibrium position, while the first method also allows you to find unstable equilibrium positions as well as the equilibrium of conservative systems. Below we will consider these two methods.

3.4.1.1. Calculation of equilibrium position by numerical solution of equilibrium equations

1. Run static and linear analysis tool.
2. Go to Options | General options tab. In group Equilibrium computation type set Solving equations as a method of defining the equilibrium position, Figure 3.8.
3. Go to Equilibrium tab and click Run computations. After the calculation the beam in the animation window will be in the equilibrium position, Figure 3.9.

Figure 3.8. Choosing a method for defining the equilibrium position

Figure 3.9. Beam in the equilibrium position
3.4.1.2. Calculation of the equilibrium position by integrating the equations of motion

1. Go to Initial conditions tab. Push Set all coordinates to zero $\times \theta$ and confirm the action.
2. Go to Options | General options tab. In Equilibrium computation type group set Integration of equations of motion as a method of defining Equilibrium position, Figure 3.10.
3. Go to Equilibrium tab and push Run computations $\bigcirc$. Observe the process in the animation window.

When calculating the equilibrium position by integrating the equations of motion, kinetic energy is used as a control of the system proximity to the equilibrium position. Integration continues until the kinetic energy drops below some small value in a given time.

In order to accelerate the convergence process, additional dissipative forces are automatically added when calculating the equilibrium position. The generalized forces of additional dissipation are proportional to the product of the mass matrix per column of velocities:

$$Q_{diss} = -\alpha M(q)\dot{q}$$  \hspace{1cm} (3.1)

Damping coefficient $\alpha$ is set on a tab Options | Parameters in parameter of additional damping field. Run the equilibrium calculation with different values of the coefficient $\alpha$, for example 0.1, 0.5 and 1. Observe the effect in the animation window. In Figure 3.10 there is a comparison of the kinetic energy drop when $\alpha = 0.1$, $\alpha = 0.5$ and $\alpha = 1$.

![Figure 3.10](image)

Figure 3.10. Kinetic energy drop at different values of the additional damping parameter
3.4.2. Calculation of natural frequencies and modes of oscillations

The eigenfrequencies of vibrations of the cantilever beam are calculated using the formula [2]:

\[ f_k = \frac{\lambda_k^2}{m_0} \sqrt{\frac{EJ}{m_0L}}, \]  
(3.2)

where \( E \) is the elasticity modulus, Pa;
\( J \) is a moment of inertia of the cross-section of the beam, \( m^4 \);
\( m_0 \) is the mass per unit length of the beam, kg/m;
\( L \) is beam length, m;
\( k \) is a frequency number;
\( \lambda \) is a parameter that is defined as follows:

\[ \lambda_1 = 1.875, \quad \lambda_2 = 4.694, \quad \lambda_k = \frac{2k - 1}{2} \pi \text{ when } k > 2. \]  
(3.3)

The amplitude functions (mode shapes) are determined with the expression:

\[ w_k(x) = K_2(\lambda_k)K_3\left(\frac{\lambda_k x}{L}\right) - K_1(\lambda_k)K_4\left(\frac{\lambda_k x}{L}\right), \]  
(3.4)

where \( K_1, K_2, K_3, K_4 \) are Krylov functions, which are defined using the following formulas:

\[ K_1(x) = \frac{1}{2}(\cosh(x) + \cos(x)), \]
\[ K_2(x) = \frac{1}{2}(\sinh(x) + \sin(x)), \]
\[ K_3(x) = \frac{1}{2}(\cosh(x) - \cos(x)), \]
\[ K_4(x) = \frac{1}{2}(\sinh(x) - \sin(x)). \]  
(3.5)

Calculate the eigenfrequencies and mode shapes of the beam in Universal Mechanism. Firstly, turn off the gravity so that it does not distort its own oscillations in the animation window. To do this, follow these steps:

1. Close model in UM Simulation program and open it in UM Input.
2. In the model tree choose Object.
3. Go to Object tab on the inspector panel on the right. In Gravity force direction group in ez field put identifier gravity_factor with 0 value, Figure 3.11.
4. Save the model and go to UM Simulation program.
Run static and linear analysis tool. Go to Frequencies/Eigenvalues tab. Push Run computations. After the calculation is completed, you will see eigenfrequencies of the model in the list on the left, Figure 3.12. To animate the mode shapes, click Run/Stop animation or double-click the left mouse button on the frequency value in the table. Click or Esc to stop the animation.

Comparison of analytical and numerical values of the cantilever beam oscillations is in Table 3.2. Comparison of the lower mode shapes of oscillations is given in Figure 3.13.
## Comparison of the beam oscillations frequency

<table>
<thead>
<tr>
<th>( k )</th>
<th>Analytical solution, Hz</th>
<th>Numerical solution, Hz</th>
<th>Relative error, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.64311</td>
<td>0.64543</td>
<td>-0.36</td>
</tr>
<tr>
<td>2</td>
<td>4.03059</td>
<td>4.06788</td>
<td>-0.93</td>
</tr>
<tr>
<td>3</td>
<td>11.2839</td>
<td>11.4467</td>
<td>-1.44</td>
</tr>
<tr>
<td>4</td>
<td>22.1165</td>
<td>22.5319</td>
<td>-1.88</td>
</tr>
<tr>
<td>5</td>
<td>36.5600</td>
<td>37.3530</td>
<td>-2.17</td>
</tr>
</tbody>
</table>

Figure 3.13. Comparison of the first two mode shapes of the cantilever beam oscillations obtained numerically (on the left) and analytically (on the right)
3.4.3. Euler’s critical load

Longitudinal compression load on the beam change the eigenfrequencies. When stretched, the natural frequency increases, when compressed it decreases. The column will remain straight for loads less than the critical load. The "critical load" is the greatest load that will not cause lateral deflection (buckling). For loads greater than the critical load, the column will deflect laterally. The critical load puts the column in a state of unstable equilibrium. A load beyond the critical load causes the column to fail by buckling. As the load is increased beyond the critical load the lateral deflections increase, until it may fail in other modes such as yielding of the material.

The critical force (Euler force) for a beam is calculated by the following formula [3]:

\[ P_{cr} = \frac{\pi^2 E J}{(\mu L)^2} \]  \hspace{1cm} (3.6)

where \( E \) is the modulus of elasticity, Pa;
\( J \) is a moment of inertia of the beam cross-section, m\(^4\);
\( \mu \) is the column effective length factor, for cantilever beam \( \mu = 2 \);
\( L \) is the length of the beam, m;

For the considered beam the critical force is

\[ P_{cr} = \frac{\pi^2 \cdot 2.1 \cdot 10^{11} \cdot 2.01 \cdot 10^{-10}}{(2 \cdot 3)^2} = 11.58 \text{ N.} \]  \hspace{1cm} (3.7)

Let us check the dependence of the first eigenfrequency of the beam on the magnitude of the compressive force. Let us apply the compressive force to the free end of the beam. To do this, follow these steps:

1. Close the model in **UM Simulation** and open it in **UM Input**.
2. Add a new identifier **Fy** to the list of model parameters and set it to 0.
3. Add new **T-Forces** into the model, Figure 3.14.

   ![Figure 3.14. Adding new T-Forces](image)

4. Set its name to **ForceY**.
5. Go to settings **T-Forces** on the inspector panel on the right. In field **Body1** select **Base0**, in field **Body2** select **Body11**, in field **Reference frame** select **Base0**. In field **Reduction point | y** set $l/4$. In field **Force | y** set -$F_y$, Figure 3.15.

6. Save model and go back to **UM Simulation** program.

![T-force ForceY](image)

Figure 3.15. T-force **ForceY**

Run **static and linear analysis** tool. Go to **Root locus** tab. In **Type of problem** select **Frequencies**. In **Identifier** field choose **Fy**. In **Limits** fields assign the limits of variation of identifier. In the upper field (initial value), enter 0, in the lower field (final value), enter a value slightly bigger than the critical force, for example, 12. In the **Count** field set 20, i.e. 20 calculations will be performed with an equal increment value of the specified identifier. Go to **Options | General options** tab. In **Equilibrium computation type** group set **Solving equations** as a method of determining the equilibrium position, Figure 3.9. Click **Run computations** button. After the calculation is completed, plot the dependence of the first natural frequency of the beam on the value of the identifier **Fy**, by clicking $\times$ in **static and linear analysis** window, Figure 3.16. The intersection of the hodograph with the x-axis corresponds to the loss of stability, Figure 3.17. A comparison of the analytical and numerical solutions shows their proximity.
Figure 3.16. Construction of root locus

Figure 3.17. Dependence of the first eigenfrequency of the beam on the magnitude of the compressive force
### 3.4.4. Large deflections

Let us consider the case of loading the cantilever beam vertical force at the free end, causing large displacement of the free end section: vertical $w$ and horizontal $u$. In [4] the analytical solution of this problem is given. Since the solution is huge, it will not be cited here. The solution for some special cases is given in the Table 3.3, where $k$ is the load parameter, $w/L$ and $u/L$ dimensionless vertical and lateral deflections.

Let us compare the analytical solution with the numerical solution in Universal Mechanism software. At first, apply a vertical force to the free end of the beam that causes large movements. To do this, follow these steps:

1. Close the model in **UM Simulation** program and load it in **UM Input**.
2. In model parameters list add a new identifier $F_z$ and set it to $k \cdot \text{YoungsModulus} \cdot \frac{J}{(\text{BeamLength} \cdot \text{BeamLength})}$, see Figure 3.18.
3. Add in the model new **T-Force ForceZ**. It will be a force at the end of the beam.
4. Go to the settings of **T-Force** in the inspector panel on the right. In **Body1** field select **Base0**, in **Body2** field select **Body11**, in **Reference frame** field put **Base0**. In **Reduction point** $y$ field set $l/4$. In **Force** $z$ field put $F_z$.
5. Save model and run **UM Simulation**.

<table>
<thead>
<tr>
<th>$k = \frac{PL^2}{EI}$</th>
<th>$w/L$</th>
<th>$u/L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>0.083</td>
<td>0.004</td>
</tr>
<tr>
<td>0.50</td>
<td>0.162</td>
<td>0.016</td>
</tr>
<tr>
<td>0.75</td>
<td>0.235</td>
<td>0.034</td>
</tr>
<tr>
<td>1</td>
<td>0.302</td>
<td>0.056</td>
</tr>
<tr>
<td>2</td>
<td>0.494</td>
<td>0.160</td>
</tr>
<tr>
<td>3</td>
<td>0.603</td>
<td>0.255</td>
</tr>
<tr>
<td>4</td>
<td>0.670</td>
<td>0.329</td>
</tr>
<tr>
<td>5</td>
<td>0.714</td>
<td>0.388</td>
</tr>
<tr>
<td>6</td>
<td>0.744</td>
<td>0.434</td>
</tr>
<tr>
<td>7</td>
<td>0.767</td>
<td>0.472</td>
</tr>
<tr>
<td>8</td>
<td>0.785</td>
<td>0.504</td>
</tr>
<tr>
<td>9</td>
<td>0.799</td>
<td>0.531</td>
</tr>
</tbody>
</table>

**Table 3.3.**

Deflections of the free end of the cantilever beam with concentrated force
How to create variables \( w/L \) and \( u/L \).

1. Open the **Wizard of variables** (Tools | Wizard of variables...).

2. Go to **Identifiers** tab, in the list of identifiers select **BeamLength**. Click \( \square \) button to create the correspondent variable.

3. Go to **Linear variables** tab, in the list of bodies select **Body11**, in **Coordinates of point in the body-fixed frame of reference**| \( y \) field put \( 0.075 \), in **Type** group choose **Coordinate**, in **Component** field choose \( Z \). Create the variable by clicking \( \square \) button. We have just created the absolute vertical deflection.

4. Now let us create the absolute lateral deflection of the same point. You have to change juts **Component** field. So, set **Component** to \( Y \) and put the variable in the container. After this the variables container should include 3 variables: **BeamLength**, \( r:z(Body11) \) и \( r:y(Body11) \).

5. Go to **Expression** tab. Let us create \( w/L \) variable:
   - add a division operator using the button \( / \);
   - use mouse to move variables \( r:z(Body11) \) and **BeamLength** in the operator fields, Figure 3.19;
   - in the field with the name of variable put \( w \) and send the variable into container, Figure 3.19.
6. Let us create the variable $u/L$:
   - add subtraction $-$ and division $/$ operators;
   - use mouse to move variables $r:y(Body11)$ and $BeamLength$ in the operator fields, Figure 3.20.
   - in the field with the name of variable put $u$ and send the variable into container.
Let us turn to the calculation of deflections. Run static and linear analysis tool. Go to Equilibrium tab. Using mouse move $w$ and $u$ variables from the Wizard of variables container into the list of variables of the static and linear analysis tool, Figure 3.21.

In the Identifier field select $k$. In the Limits field assign the edges of the identifier change. In the upper field (initial value) enter 0.25, in the lower field (final value) enter 9. In the Count field enter 36, which corresponds to the increment step of the load parameter 0.25. Click Run computations. After the calculation is completed, plot the dependence of the free end of the beam movements on the load parameter $k$. To do this, drag the calculated variables from the list of variables to the graphic windows. The graphs are shown in Figure 3.22 and Figure 3.23.
Figure 3.22. Vertical movement

Figure 3.23. Horizontal movement
Comparison of analytical and numerical results is shown in Table 3.4 below.

### Table 3.4

<table>
<thead>
<tr>
<th>$k = \frac{PL^2}{EJ}$</th>
<th>Analytical solution</th>
<th>Numerical solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$w/L$</td>
<td>$u/L$</td>
</tr>
<tr>
<td>0.25</td>
<td>0.083</td>
<td>0.004</td>
</tr>
<tr>
<td>0.50</td>
<td>0.162</td>
<td>0.016</td>
</tr>
<tr>
<td>0.75</td>
<td>0.235</td>
<td>0.034</td>
</tr>
<tr>
<td>1</td>
<td>0.302</td>
<td>0.056</td>
</tr>
<tr>
<td>2</td>
<td>0.494</td>
<td>0.160</td>
</tr>
<tr>
<td>3</td>
<td>0.603</td>
<td>0.255</td>
</tr>
<tr>
<td>4</td>
<td>0.670</td>
<td>0.329</td>
</tr>
<tr>
<td>5</td>
<td>0.714</td>
<td>0.388</td>
</tr>
<tr>
<td>6</td>
<td>0.744</td>
<td>0.434</td>
</tr>
<tr>
<td>7</td>
<td>0.767</td>
<td>0.472</td>
</tr>
<tr>
<td>8</td>
<td>0.785</td>
<td>0.504</td>
</tr>
<tr>
<td>9</td>
<td>0.799</td>
<td>0.531</td>
</tr>
</tbody>
</table>
4. Subsequent studying of Universal Mechanism

You have come through two examples of dynamical systems (pendulum and sprung body) and have seen the basic tools and features of the UM Base version.

The Getting Started series includes other manuals that devoted to the rest modules of the Universal Mechanism. Here they are:

- Getting Started: simulation of road vehicles;
- Getting Started: railway vehicle dynamics;
- Getting Started: Matlab/Simulink interface;
- Getting Started: scanning and optimization module;
- Getting Started: simulation of flexible bodies with UM FEM;
- Getting Started: durability analysis.

Library of simple models: how to…

The UM User’s Manual includes 07_UM_Simulation_Examples.pdf, which is devoted to consideration of simple models that show you how to create/model various graphical elements, joints and force elements. Studying these examples helps you familiarize yourself with basics of Universal Mechanism and approaches for simulation of objects of different kind. The library of models is in the \{UM Data\}\SAMPLES\LIBRARY directory.

The 07_UM_Simulation_Examples.pdf you can find in the \{UM Data\}\MANUAL directory or download using the following link:

5. References


[2] Euler–Bernoulli beam theory,
   https://en.wikipedia.org/wiki/Euler%E2%80%93Bernoulli_beam_theory.
