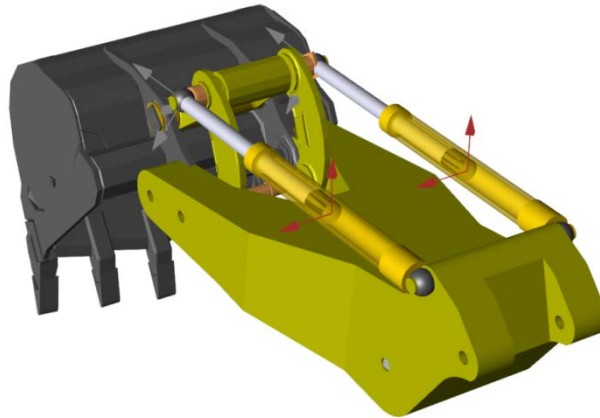


## Fact Sheet

### 3D View

Using multidimensional information in a 3D environment is a great advantage to working with a simple 2D view. Beside the realistic display of geometries, movements and further simulation results there is the possibility to model 3D graphics interactively. SimulationX allows displaying and animating multi-body systems (MBS) models in one or more 3D views as part of several libraries, such as *MBS*, *Animation Bodies* and *Planar Mechanics*. Its three dimensional visualization is characterized not only by quality but also by a simple usage and a variety of possibilities to interact and manipulate.

The key interactions in a 3D system are navigation, selection and manipulation. Using mouse and keyboard, requirements and expectations on navigation and selection are the same across the most 3D systems. In contrast, the requirements regarding manipulation are very different with respect to complexity and accuracy. CAD systems offer a lot of different tools to create and modify objects with high precision, while 3D modeling applications allow the user to modify the mesh itself.



The main aspect of SimulationX is the generation, simulation and analysis of models. The modeling part can be divided into creating objects, connecting objects and specifying parameters for objects. The 3D view offers a direct way to work with MBS models providing the same accuracy which is offered by the conventional way of modeling. Handling MBS models in this way is simpler, easier to understand and faster to accomplish.

This document presents the concepts used to enhance the 3D environment in SimulationX, focusing on manipulation. The described interaction methods allow the user to perform tasks inside the 3D view. The advantage of this approach is the direct manipulation of objects combined with real time feedback, which makes it easier for the user to validate the results of the current task, and plan the next one.

- *Improving the modeling process of MBS systems*
- *Supporting interaction tasks in a 3D environment*
- *Getting real time feedback*
- *Making common modelling tasks faster and easier*

## Create

The creation of elements is very easy in SimulationX. The user just drops the desired element out of the library tree into the diagram view or, in case of an MBS element, into the 3D view. When placed in the workspace of SimulationX the (provisional) position P of an object is defined implicitly. Depending on the type of object the generation can be enhanced by a more or less complex construction process. In this process e.g. dimension parameters can be defined using the mouse. Each parameter is constructed in a fixed direction, and can be set by a simple mouse move. This operation will be continued until all parameters are defined.

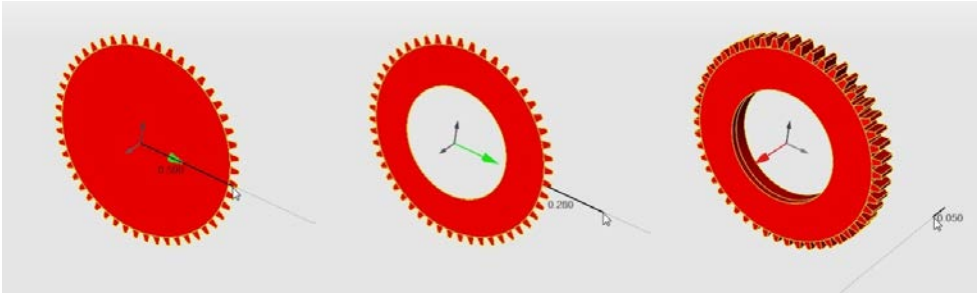


Figure 1: Construction of a tooth wheel from left to right: outer radius, inner radius, length

## Transform

The position of an object consists of a displacement and an orientation. The displacement is normally described by 3 Cartesian coordinates (an alternative would be spherical coordinates). The orientation is fully described by an orthonormal matrix R, which can be interpreted as the result of one or more rotations. SimulationX offers different sequences of rotation axes in order to define the rotation in a constructive way, for instance by means of Euler angles.

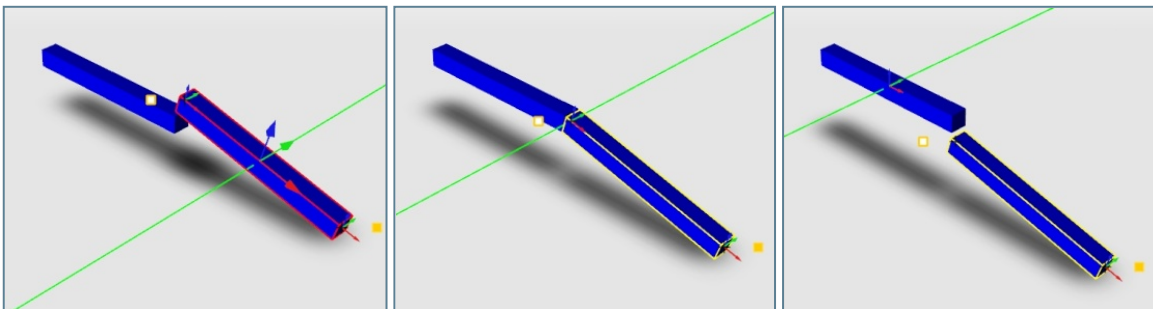


Figure 2: Rotation of object around different coordinate axes

In order to modify the position of an object the user can overwrite  $x$  and  $R$  by simply entering the respective new values in the property dialogue. More comfortably, he can use the mouse to define the change of the position, i.e. a movement  $T = \{y, S\}$  of the object, where again  $\{y, S\}$  is a pair of translation and rotation. To define them interactively with the mouse, constraints have to be used to keep control over the transformation. While different coordinate systems can be used as a reference frame for both translation and rotation, the two dimensional input values are mapped differently to the components of  $y$  and  $S$ . Translations can be restricted to coordinate axes or planes. For rotations, the input is interpreted as an angle around a specified axis.

## Connect

In order to simulate a model, objects within that model have to be connected in a certain way. In a MBS model, these connections form a kinematic chain, dividing objects in successors and predecessors for each element in that chain.

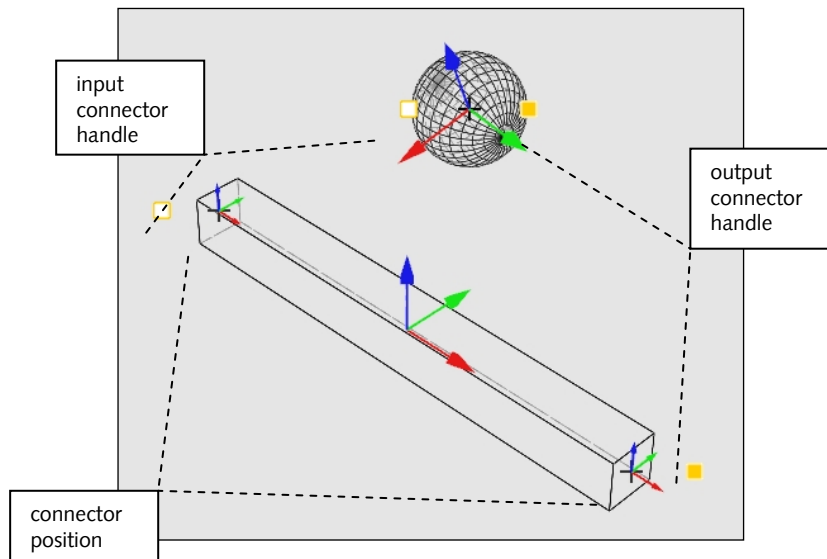


Figure 3: Layout of MBS objects

An MBS object contains several connectors, each with a specified position on the object. They are displayed when the object is selected, in form of a small coordinate system and a handle. Like in the structure view, there are input and output connectors. For the manipulation, these connectors come with certain roles.

First of all, they can be used to move the elements they belong to around the scene. Second, connectors can be used to connect objects with each other, simply by grabbing an object at one of its connectors, and dropping it on an appropriate connector of another object. Third, the positions of the connectors specify an initial transformation when objects are being connected. The object that has been moved is transformed to match the position and orientation of the two connectors.

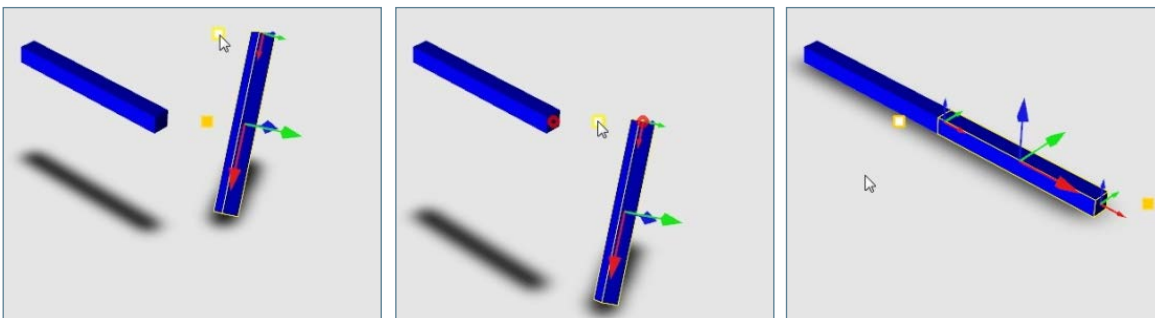


Figure 4: Left: select a connector, middle: drag it onto another one, right: result.

To prevent the user from accidentally cutting connections, they are indicated when a participating connector is translated. When the user stops the manipulation within a certain distance, the previous model state is kept. By pressing a predefined key, it is possible to transform a connector keeping its connection, thus creating a transformation between connectors.

With these rules it is possible to transform connected objects according to the dependencies of the kinematic chain, and to manipulate the structure of the chain itself.

## Visualize

The 3D visualization offers a range of advanced rendering techniques. Supported features include:

- hidden line rendering
- different shading models
- edge visualization for internal and imported geometries
- reflection, shadow planes and shadow maps

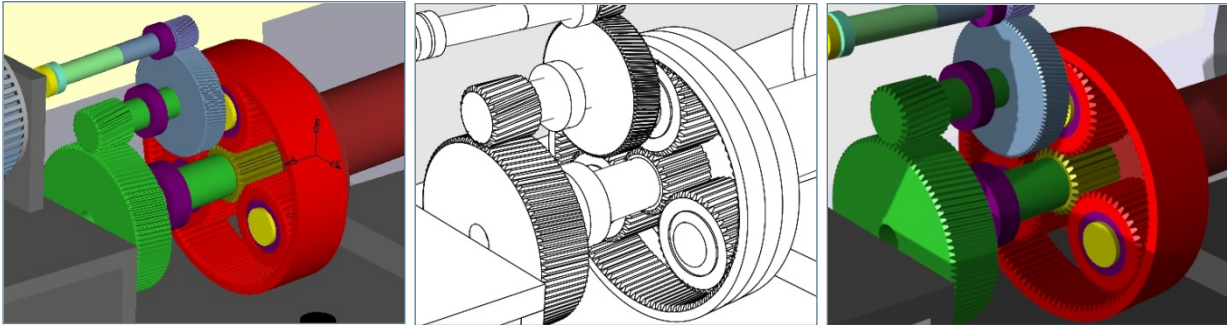


Figure 5: Visualization enhancements, left: 3D output in SimulationX 3.5, middle and left: render results using different techniques of the 3D view