

Transform a Vector Using a Rotation

base model: models/ForceVector.mdl

final model: models/ForceVector2.mdl

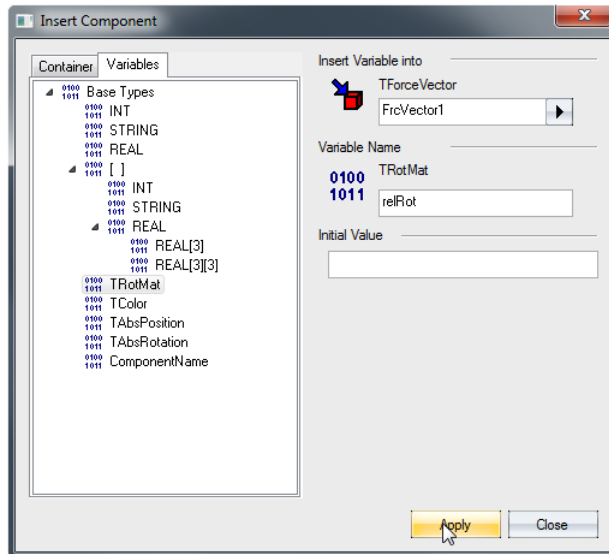
Explanation

- ▶ A force should be applied, which is constant in the global coordinate system `Ground.BFR`
- ▶ If the force is applied in a Frame, which is moving with respect to `Ground.BFR`, the vector-valued force has to be transformed from `Ground.BFR` to this Frame
- ▶ Mathematically speaking a coordinate transformation of the force vector has to be done
- ▶ Therefore a rotation matrix of the relative rotation of the Frame with respect to `Ground.BFR` will be used
- ▶ `alaska` has built-in functionality for the calculation and application of rotation matrices
- ▶ There is also a particular data type or variable type for a rotation matrix
- ▶ The variable type is `TRotMat`

Variable of type *TRotMat*

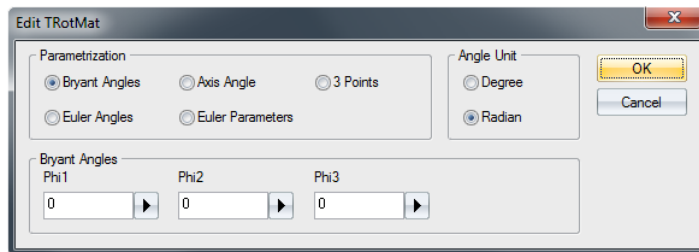
Open the base model `ForceVector.mdl` in alaska/ModellerStudio

- 1 We want to insert a variable of type *TRotMat* to `FrcVector1` and name it "relRot"
- 2 To do so: Select `FrcVector1` within the "Model Tree" and press F6
- 3 In the dialog switch to the second tab Variables
- 4 Select *TRotMat* and enter "relRot" at the "Variable Name" box as can be seen on the next screenshot, then click Apply

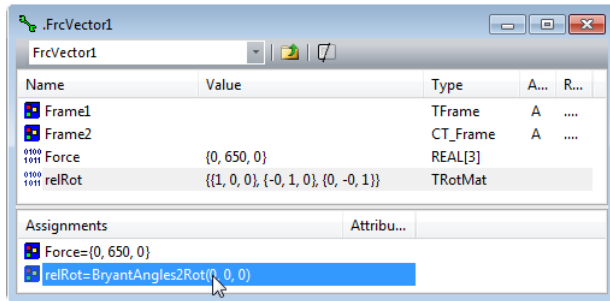


Relative rotation

- ▶ Now we want to use the built-in functionality to calculate the relative rotation
- 5 Double-click `relRot` within `FrcVector1`
- 6 The default editor for rotations will open

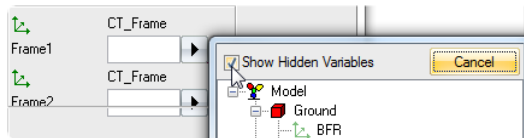


- 7 Simply click on `OK`, as we can not use this dialog
- 8 Now a statement is assigned to the `relRot` variable, which can be seen in the lower part of the "Component View" window

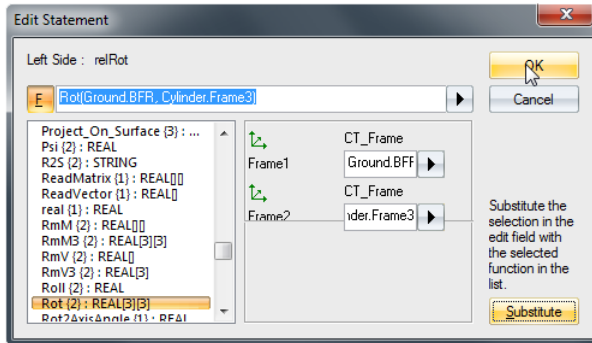


- 9 Double-click on the assignment as can be seen in the above screenshot
- 10 Now the general editor for assignments/statements is opened
 - 1 Delete the old assignment "BryantAngles2Rot(0, 0, 0)"
 - 2 Click on **E** to open the list with built-in functions
 - 3 Navigate to the function "Rot" and click on it
 - 4 Now two boxes for the arguments "Frame1" and "Frame2" will appear
 - 5 Select Ground.BFR as "Frame1" and Cylinder.Frame3 as "Frame2"

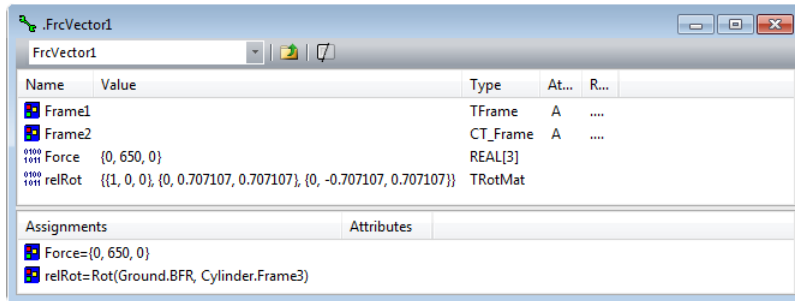
Do not forget to check **Show Hidden Variables** to be able to select `Ground.BFR`



- 6 To accept the function with its arguments click on **Substitute**
- 7 If the assignment is showing in the top box of the editor as in the next screenshot, hit **OK**



- ⑪ Now the Assignment for *relRot* is defined and the corresponding value will be calculated in any task
- ⑫ If you run the *Assembly* the value should read:

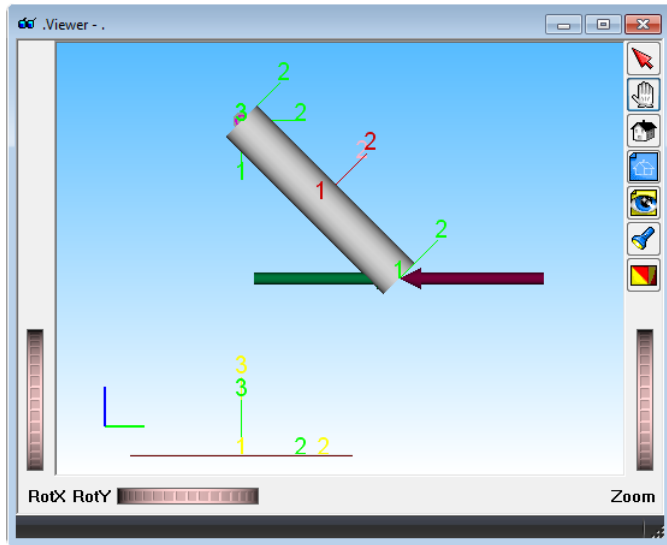


Matrix-vector multiplication

- ▶ To transform the vector $\{0, 650, 0\}$ from the Frame `Ground.BFR` to the Frame `Cylinder.Frame` we have to multiply the relative rotation to the left side of the vector (a simple matrix-vector multiplication)
- ▶ again there is built-in functionality in alaska
- ▶ The function to multiply a 3×3 matrix with a vector of length 3 is named "`M3mV3`"
- ▶ Instead of using the function editor, assignments may also be written manually:
 - 1 Double-click on `Force` within `FrcVector1`
 - 2 Change the assignment to
`M3mV3(relRot, {0, 650, 0});`

Check

- ① Double-click on `Viewer` at the “Model Tree”
 - ② Select `Batch` at the “Task Tree” and click on run
- ▶ The pendulum will oscillate due to its initial angle, gravitation and the applied force
 - ▶ **Note:** The force-vector will always stay parallel to the global y-axis



Next tutorial

Next tutorial: [Insert a Spring-Damper](#)