

SIMPACK Tips and Tricks - Expressions

SIMPACK 8.5 introduced the modeling element 'Expressions'. They are a very versatile element which can be used for many different applications.

Expressions can be seen as functions which process multiple inputs and return them as a single output. The inputs come from measurements which are performed during the simulation, and the Expression returns the result. They can be used to define a force, marker position, time excitation, y-output value or Result Element. They can also be used to return a measurement value. Similar to Substitution Variables it is possible to generate complex formulas using different functions all within one Expression. The difference however is that Expressions are 'dynamic' within the simulation and are recalculated during the integration. Substitution Variables on the other hand are calculated when the simulation is started and then remain constant.

Expressions retrieve their input data from SIMPACK Access Functions, e.g. the function JOINTCF() returns the Joint forces, or FORCEOV() returns the Output Values from a Force Element.

WHERE CAN EXPRESSIONS BE USED?

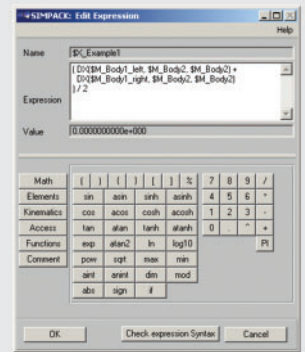
- Force Elements 50 and 51 read-in Expressions and return either a force or moment. Example models are available for both these Force Elements. Force Elements 5 and 43 can also be used with Expressions.
- Time Excitation 12 reads-in one or more Expressions, allowing the motion of a Moved Marker to be easily controlled.
- The Control-Sensor 232 returns the Expression result within a control loop.
- The Y-Output Element 19, which is driven by an Expression, is used within a co-simulation.
- The new modeling element 'Result Element' is used to generate 'General Plots' result data via Expressions and will be available in the next customer release of SIMPACK. See the article on Result Elements.

User Routines can often be replaced with an Expression when either mod-

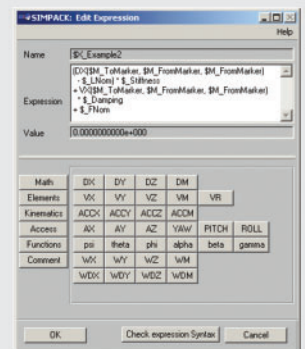
eling Force Elements or Time Excitations. They are even comparable in terms of speed, as they are compiled as a pre-processing step.

THREE EXAMPLES

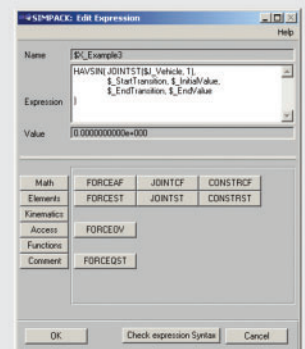
The first example calculates the distance in x between two sets of Markers and returns the mean value: „(DX(\$M_Body1_left, \$M_Body2, \$M_Body2) + DX(\$M_Body1_right, \$M_Body2, \$M_Body2)) / 2“. The calculation is performed from the function DX() which has the same parameters as a Sensor: TO-Marker, FROM-Marker und Reference Marker (see „Tips & Tricks“ from the last SIMPACK News). The second example is a little bit more complicated. The Expression calculates the force from a linear stiffness and damping, whilst also considering a nominal force and length. „(DX(\$M_ToMarker, \$M_FromMarker, \$M_FromMarker) - \$LNom) * \$Stiffness + VX(\$M_ToMarker, \$M_FromMarker, \$M_FromMarker) * \$Damping + \$FNom“ (see picture 2). The Force Element parameters come from Substitution Variables. When used with Force Element 51 this Force Element will behave in exactly the same way as Force Element 04 (Spring/Damper PtP)! All good things come in threes. The last example shows a distance dependent excitation with smoothing transitional regions between the initial and final positions. All the parameters are entered as Substitution Variables: „HAVSIN(JOINTST(\$J_Vehicle, 1), \$StartTransition, \$InitialValue, \$EndTransition, \$EndValue)“ (see picture 3). JOINTST() reads-in a Joint state, for example here the position of a vehicle along the track. HAVSIN() is used to create a smoothed transitional region between the \$InitialValue and \$EndValue. If the Joint state is smaller than \$StartTransition then the Expression returns \$InitialValue. If it is greater than \$EndTransition then \$EndValue will be returned. The returned value will be smoothed in-between \$StartTransition and \$EndTransition, ensuring that no discontinuities occur which could cause problems for the integrator.



Example 1



Example 2



Example 3