

What is new in SAM 8.3?

Dynamic Points

The possibility to construct points via the intersection of lines and even the intersection of centerlines has become dynamic. When moving the objects used in the creation of a point, the xy coordinates of that point are adapted automatically. That holds for the situation that an object is move in the reference position, but also when performing an animation.

In this way one can for example create point p1 as the intersection of beam 3 and beam 6, which constitutes the momentary center of rotation of the triangle formed by the triangular coupler. When the mechanism is animated and moves from one position to the next, the location of point p1 is adapted automatically.



The point p1 is created as an intersection of the two (infinite) lines corresponding to the beam elements 3 and 6. The point is dynamic an moves during an animation.

Distance

The postprocessing module (Results > User Defined ...) has been extended with a new function: distance(object1,object2), in which an object can be either a node, element, point or shape.

r define	d results									
Plot	t Name	For	rmula						Unit	
\[\] \[\[\] \[\] \[\] \[\[\] \[\] \[\] \[\[\] \[\[\] \[\[\] \[\[\[\[e4p1	dis	tance(e4,p1)						mm	
Add formula						Inse	rt formula	Delete formula		
Availabl	e function:	s: select to i	nsert					Available SAM variables		
cos	sin	tan	cot	min	max	mean	ddt	pi	Component Nr Result/Property	
abs	sign	sqrt	log	int	first	last	intf		Node 💌 1 💌 X 💌	
exp	asin	acos	atan	rms					n1 X	
acot	sinh	cosh	tanh						114_24	
coth	gtz	distance								
wample							1	1		
xample	25:									
- 2.3 *	' sin(X) +Y^	2							· · · · · · · · · · · · · · · · · · ·	
gtz(sin	(t) +1.234	- 1.123E-3))							
distance log(n2	23,545 Xx + 2.3) + 1.2345 m3 Vahs) +	- mvVariable							
sin(pi/1	180*e3_A	a)	ing randole							
n1 v /	101001	-ha*a?	n 946							
									OK Cancel	
									UK Calice	

Definition of the distance between element e4 and point p1.

The use of this feature is manyfold:

1. Plot the distance between two objects as function of time. In the previous example it might be useful to plot the distance between the driving cylinder (element e4) and the dynamic point p1 in order to monitor if that distance is always positive and does not reach zero, since in that situation the mechanism would be blocked.



Distance between element e4 and point p1 as function of time is plotted next to the mechanism

2. During an optimization, the xy-location of a node can be an optimization parameter. Using the distance function one can define a penalty function for the optimization in order to constrain the xy-location of that node to any polygon



Optimization in which node n6 is a design parameter but should be constrained to the area of the polygon (shape s1).

d results											
Name	F	xmula stance(n6,s1) st(n6s1) 000°gtz(temp)°temp^2									
n6s1 distance(n6,s1)											
temp first(n6s1)											
Image: penalty 1000*gtz(temp)*temp^2											
	Add form	nula			Inser	t formula		Delete formula			
e functions	: select to	o insert						Available SAM variables			
sin	tan	cot	min	max	mean	ddt	pi	Component Nr Result/F	Property		
sign	sqrt	log	int	first	last	intf		Node 💌 1 💌 X	-		
asin	acos	atan	rms					n1 X			
sinh	cosh	tanh									
gtz	distanc	:e									
s:			,			,					
*pi) sin(X)+Y^ (t) +1.234 e(s23,s45)	2 - 1.123E- + 1.234	·3) 5							^		
	Name n6s1 temp penalty functions sin sign asin sinh gtz s: pin(X) +Y ∧ tr y+1.234	Name F n6s1 d temp fi penalty 1 Add form functions: select to sin tan sign sqrt asin acos sinh cosh sin gtz jt jt jt jt jt jt	Name Formula n6s1 distance(n6,s1) temp first(n6s1) penalty 1000*gtz(temp)* Add formula functions: select to insert sin to atan sinh cosh tan gtz distance s: p) pin(X)+Y^2 () +1.234-1.128E-3)	Name Formula n6s1 distance(n6,s1) temp first(n6s1) penalty 1000*gtz(temp)*temp^2 Add formula functions: select to insert inin tan cot isin cos atan igtz distance functions: select to insert p) in(X)+Y^22 j+1.224-1.122E-3)	Name Formula n6s1 distance(n6,s1) temp first(n6s1) penalty 1000"gtz(temp)#temp^2 Add formula functions: select to insert sin tan sin acos sinh cosh gtz distance st phy	Name Formula n6s1 distance(n6,s1) temp frst(n6s1) penalty 1000*gtz(temp)*temp^2 Add formula Inser functions: select to insert ini sin tan sign sqt sinh cos gtd distance	Name Formula n6s1 distance(n6,s1) temp first(n6s1) penalty 1000*gtz(temp)*temp^2 Add formula Insert formula functions: select to insert int sin tan sign sqrt sinh cosh gtd distance	Name Formula n6s1 distance(n6,s1) temp first(n6s1) penalty 1000*gtz(temp)*temp^2 Add formula Insert formula functions: select to insert int sin tan sinh cosh gtd distance	Name Formula n6s1 distance(n6,s1) temp first(n6s1) penalty 1000**gtz(temp)*temp^2 Add formula Delete formula Delete formula functions: select to insert Available SAM variables functions: select to insert int first last intf int first last int first int first last intf int first last int first i		

Definition of the penalty function to constrain node n6 to the polygon s1. As soon as node n6 lies outside of the polygon in the initial position of the mechanism, the penalty is larger than 0.

3. Collision detection between objects

The distance function can be used to detect collision between objects and depending on the type of objects even the penetration depth (or minimum retraction distance to avoid collision). In the following example the blue polygon is a stationary shape s1, whereas the yellow triangle (shape s7) is attached to the coupler. In the plot one sees the variable s7s1, which corresponds to the distance between the two shapes and the variable p23s1, which corresponds to the distance between point p23 on the triangle and the blue polygon. In case of two colliding shapes the distance is set to zero, whereas in the case of a point or node moving into a shape both the positive and negative distance (negative distance = penetration depth) is shown in the plot.



Example of collision detection via use of the distance function.

Copy/Paste (Ctrl-C/Ctrl-V) for graphics/shapes

One can now use Ctrl-C and Ctrl-V to copy and then paste any shape.