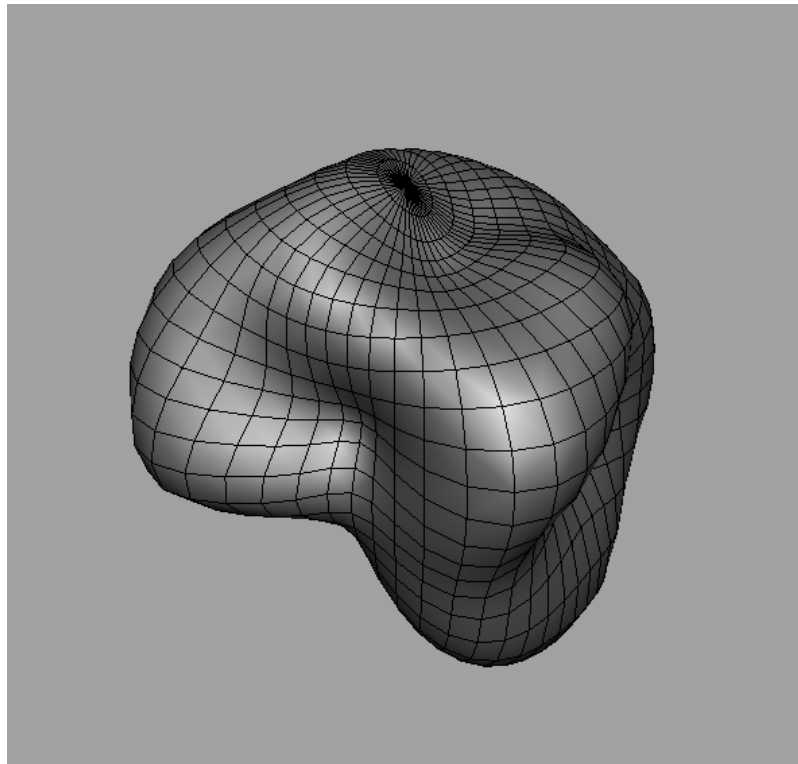


1 Wobble - A Noise-Based Deformer

The Wobble-Plug-In supplies a deformer, which allows you to apply animated irregularities - or “wobbling” - to an object. It uses some algorithms based on Perlin noise.



Attribute (Short Name)	Type	Description	Default
noiseType (nt)	- simple noise - turbulence - fbm	<p>simple noise: basic perlin noise</p> <p>turbulence: turbulence function $noise(f) /noise(f)$</p> <p>fbm: fractional brownian motion $1/f \cdot noise(f)$</p> <p>Notice that turbulence and fbm require fairly highly subdivided geometry because they contain higher frequencies (turbulence: 3 octaves, fbm: 5 octaves). In case of too low subdivision a polygon mesh will get very chaotic, NURBS and SUBD surfaces will simply filter out the higher frequencies.</p>	simple noise

Attribute (Short Name)	Type	Description	Default
periodic (per)	boolean	if true, the noise pattern repeats seamlessly in intervals given by the “Periods Space” and “Periods Time” attributes	false
strength (st)	distance	the amount/amplitude of displacement.	1.0
spaceFrequency (sfr)	float float float	spatial noise frequency for x, y and z. This attribute only uses values from 0 to 1 to set up the relations between the three dimensions. Use the frequency-scale-attribute to adjust the overall value.	0.88 0.88 0.88
spaceOffset (sof)	float float float	This allows to shift the noise without having to be in deformer space mode and independent from the deformer handle's transformation	
freqScale (sfs)	double	an overall scaling parameter for the spatial frequency	1.0
timeFrequency (tfr)	double	animation frequency.	1.0
periodsSpace (ps)	int int int	interval after which noise pattern repeats (x, y and z).	3 3 3
periodsTime (pt)	int	interval, after which noise animation repeats. Depends on the input of the time attribute, which is by default connected to the time node, so this attribute is related to seconds. Mind the frame rate to create seamlessly repeating animations.	3
deformMode (df)	enum: - planar - cylindrical - spherical - normal - random vector	planar: the deformation is applied directional along the deformer handle's up-axis. Rotate the deformer handle to change the deformation direction. cylindrical: the deformation is applied cylindrical around the deformer handle's up-axis.	planar

<i>Attribute (Short Name)</i>	<i>Type</i>	<i>Description</i>	<i>Default</i>
		<p>spherical: the deformation is applied in a spherical manner along the vector from the deformer handle's origin to each point</p> <p>normal: the deformation is applied along the surface normal at each point. Very slow with NURBS. Not supported for subdivision surfaces.</p> <p>random vector deformation is applied in an random direction (NOT YET IMPLEMENTED)</p>	
deformationSpace (dsp)	enum: - deformer - world	<p>deformer: the noise is in deformer space. All transformations on the deformer handle affect the noise. You can parent the deformer under the deformed object to get kind of an “object space” mode.</p> <p>world: the noise is in world space. Transforming the deformed object will push it through the noise.</p>	world
deformDirection (dir)	enum: -average -in -out	determines how the deformation is applied tho the surface	average
falloffType (fot)	enum: -infinite -cylindrical -spherical -cubical	<p>none: the deformation is applied equally everywhere</p> <p>cylindrical: the deformation is applied only inside in a cylindrical volume. Thus it is only restricted in two axes</p> <p>spherical: the deformation is applied only inside in a spherical volume.</p>	infinite

Attribute (Short Name)	Type	Description	Default
		<p>cubical: the deformation is applied only inside in a cubical volume.</p> <p>Deformation can fade over a certain range towards the borders of the sphere. Adjust this using the “falloff amount”-attribute.</p>	
falloffDistance (fod)	distance	<p>if in cylindrical or spherical mode: radius</p> <p>if in cubical mode: half side-length</p>	5.0
falloffAmount (foa)	percent(float)	falloff range inside the sphere. This Attribute is measured in percent of the falloff distance	20.0
falloffInterpolation (foi)	enum: -linear -smooth -ramp	the way the fading over the falloff range gets interpolated	linear
inverseFalloff (ifo)	Boolean	Falloff works inverse	False (off)
falloffRamp (for)	Ramp	If falloffInterpolation is “ramp”, the the interpolation is determined by this value (scalar) ramp	-
time (t)	time	this attribute is by default connected to the time1 node to get the correct time information during playback. In most cases this is how you want it to be, however actually you can plug in everything you want here to get some non-linear animation of the deformation.	connected to default time node