



**parameters or genes:** Octopus can connect to multiple GenePool and NumberSlider components. Like for any other components, hold shift to connect additional wires, hold Ctrl to remove single connections. Octopus will explore possible values of the connected Sliders according to their individual range settings [0 to 100, 5 to 9, or 3.1 to 62.3, etc.]

**parametric model or mapping:** Takes some input parameters and does some calculations / analysis / generation .. How long this step takes is important for the overall performance of the search, since a lot of evaluations with different parameter values have to be performed.

### objectives and phenotype:

**Solution Mesh:** optionally, a 3D mesh representation of the solution can be fed into Octopus for visual assessment. Meshes with high number of faces will increase the memory usage of Octopus and increase the filesize of the definition when saved.

**Numeric objective values [Number parameter]:** The fitness values of a solution. Minimum two, maximum is theoretically unlimited.

**Textual objectives description [Text parameter]:** Simple short names describing the objective dimensions

**Boolean hard constraints [Boolean parameter]:** Optional, a boolean parameter can be connected. Octopus expects a 'true' value for every valid solution, otherwise the solution is thrown away.



- 1 Main viewport / viewcube. Opaque cubes indicate the non-dominated pareto-front, transparent cubes are dominated solutions still belonging to the Elite. Transparent yellow cubes are elite-solutions from previous generations [history], the more transparent the older. Transparent yellow spheres indicate a simple marking. Those marked solutions are shown all the time. On the coordinate axes, and, if used, the additional color- and size-dimensions, a minimum and maximum value is shown for the respective objective dimension - corresponding to the set which is currently selected to fit into the viewcube.
- 2 Context menu when left-clicking a solution. **'Reinstate Solution'** restores the solutions' parameters [genes] in the Grasshopper setup and recalculates it. **'Mark'** adds a yellow sphere around the solution to keep it alive and visible until unmarked, also a yellow polyline is added at the parameter-graph [item 9]. **'Mark Preferred (Obj)'** adds a user preference at this region in the objective space, meaning the weighting of objectives is said to be preferred. Totally different parameters of two solutions can lead to similar objective values, so another level of preference articulation is needed to favor parametrically [genetically] similar solutions [and hence, often formally similar]: **'Mark Preferred (Gen)'**, which introduces an additional objective dimension, no matter how many genetic preferences are set. 'Mark Preferred (Phen)' is of no function at the moment. **'Toggle Show Mesh'** toggles between the cube- and mesh-view of a single solution. **'Delete'** deletes the solution.
- 3 History Slider, which allows you to scroll through the history of the search process. All history except the last generation can be deleted with the 'X' - button. The search can be resumed from any point in history, the generations of the new [parallel] timeline are just added at the end of the history. Also, markings and preferences can be done in history.
- 4 Process control: **'Start'** starts the process with 2 x PopulationSize of random solutions in the beginning, or resumes a process that has been stopped. **'Start with Presets'** can be used to incorporate the Slider-Setup currently in Grasshopper. When a search is started for the first time, 2 x PopulationsSize solutions are generated from the current Slider-Setup by using the current settings for mutation [see 5]. When a search is resumed, just one solution with the current slider setup is added to the pool. Both start buttons will open a little additional window that runs in a separate thread to stop heavy calculations anytime. **'Stop'** stops a search. Stopped searches are always resumable. **'Reset'** clears all solutions, preferences, history, etc., but not the settings for Octopus itself.
- 5 Algorithm Settings: **'Elitism'** gives the percentage of new solutions that are bred out of the Elite instead of the entire pool. When set high, more local optimization is performed. **'Mut. Probability'** is the probability of each parameter /gene to become mutated with the **'Mutation Rate'**. A low Mutation Rate means little changes to the parameters' values, a high rate means big changes. This can be tested with one of the commands in the menu Operators - TestMutation. **'Crossover Rate'** is the probability of two subsequently generated solutions to exchange parameter values.  
**'Population Size'** is the number of solutions per generation. The Elite size is set accordingly, so a total of 2 x PopulationSize number of solutions are in each generations' pool. This size should be set according to the complexity of the problem, since a lot of solutions at the same time can maintain a lot of different alternatives. **'Max. Generations'** is set to zero by default, meaning there is no end to the search. Otherwise Octopus will stop after this number of generations. **'Record Interval'** is the interval of generations in which a history record is stored. Can reduce the memory footprint for long searches. **'Save Interval'** gives the interval of generations after which the Grasshopper file is saved to prevent data loss when Rhino crashes during search for whatever reason. **'Max. Eval Time [ms]'** - if a solution takes longer than this to compute, it is added to a special collection of solutions that can be debugged afterwards, by reinstating them in the 'Troubleshooting' tab. Can be used to track special unwanted situations for specific parameter values. **'Single Steps'** lets the search stop after each generation.  
**'Minim. Rhino on Start'** minimizes Rhino and Grasshopper when a search is started, because performance of the evaluation can go up 20 times when the screen updates dont have to be performed. However this might not work for non-english installations of Grasshopper, then you should minimize manually. **'Constraints: Fill Random'** is a setting only relevant when hard constraints are specified by connecting Octopus Objectives to a boolean-component. When fill random is enabled, the search stays within the first generation until it has found enough valid random solutions to fill the first generation up to 2 x PopulationSize. Otherwise, just a minimum number of 6 solutions is required and the pool will eventually fill up with valid solutions over time. Enabling is recommended to have a good spread of different solutions from the beginning on. 'Diversify Parameters' introduces an additional objective dimension which favors solutions very different from others in their parameter-configuration (gene-space). Eventually, the Checkbox has to be unchecked and checked to work. **'Record Solution Meshes'**: when checked, the mesh-representation is also stored for history solution which greatly increases memory usage. By default only the last generation stores solution meshes.  
**'SPEA-2 or HypE reduction'** lets you choose between two different strategies of how a pareto non-dominated front should be truncated to fit the archive size when it is too big. **'Polynomial, Alternative Polynomial, HypE, or Custom Mutation'** lets you choose between different mutation strategies.

- 6** Display Settings: '**ParetoFront, Elite, History**': The row of Checkboxes determines if a set is shown at all. The row of Radio-Buttons determines the scale of the scene by fitting the respective set into the viewcube. Additionally, history has a slider for how many history solutions to show. '**Meshes**' means the optional solutions meshes that can be plugged into Octopus. When supplied, this checkbox is a global toggle between cube- and mesh-view of the solutions. The slider determines the scale of the meshes in the viewport. '**Scale**' controls the scale of the solutions' cubes.
- 7** Hypervolume Graph: The Hypervolume is a mathematical measure for the spread of solutions and is used by the algorithm, but the implementation here as a history graph is experimental.
- 8** Statistics during the process
- 9** Genetic distance graph: Each row represents a parameter [gene], where the corners of the polylines represent values of that parameter. Each solution shown in the main viewport has a polyline in the genetic distance graph. This can give an overview of the convergence of a search.
- 10** List of objectives by their name and in the order of how they are supplied to Octopus in Grasshopper. The 'Use' column can be used to neglect objectives within the algorithm. This way they are just calculated and can be shown, but are not considered in comparing solutions to each other. 'Spread' has no function at the moment. The 'Axes' group is used to assign objectives to be shown on different axes [either the three spatial axes, the coloring, or the sizing of solutions]. One objective can be shown on multiple axes, but of course one axis can only show one objective. At 'max:', a maximum value for the respective objective dimension can be set. The main viewport is then scaled to this maximum. For all future evaluations, solutions exceeding this maximum are thrown away. The button 'Del. outer solutions' deletes all existing solutions exceeding a maximum.
- 11** Convergence Graphs: One graph for each objective dimension, showing the upper- and lower-bounds of the pareto-front (dark gray) and the Elite (light gray, background) for the number of history solutions specified in the Display Settings [6].

## View

<input checked="" type="checkbox"/>	ShowCoordinateSystem
<input checked="" type="checkbox"/>	ShowGridLines
<input type="checkbox"/>	ShowCameraInfo
<input type="checkbox"/>	ShowTriangleCountInfo
<input type="checkbox"/>	ShowFieldOfView
<input type="checkbox"/>	ShowFrameRate
<input checked="" type="checkbox"/>	Enable Clipping Planes
<input checked="" type="checkbox"/>	Enable Billboard Labels
<input type="checkbox"/>	Show Delaunay Front Mesh
<input type="checkbox"/>	Show 2D Front Line
<input type="checkbox"/>	Show Genetic Relations
<input type="checkbox"/>	Show All Genetic Relations
<input checked="" type="checkbox"/>	Show Context Menu On Click
<input type="checkbox"/>	Zoom On Click
<input type="checkbox"/>	Front Mesh Smoothing Level ▶
<input type="checkbox"/>	Title ▶
<input type="checkbox"/>	SubTitle ▶
<input type="checkbox"/>	Info

## Cam

<input type="checkbox"/>	Orthographic
<input type="checkbox"/>	InfiniteSpin
<input checked="" type="checkbox"/>	IsInertiaEnabled
<input type="checkbox"/>	RotationSensitivity ▶
<input type="checkbox"/>	ZoomSensitivity ▶
<input type="checkbox"/>	CameraInertiaFactor ▶
<input type="checkbox"/>	Reset

## Meshes

means the optional solutions meshes

<input type="checkbox"/>	Recalc Current Eval Pool
<input type="checkbox"/>	Recalc Front
<input type="checkbox"/>	Toggle All Individual Visibility
<input type="checkbox"/>	Toggle Visible Individuals' Visibility
<input type="checkbox"/>	Set All Individual Visibility False
<input type="checkbox"/>	Set All Individual Visibility True
<input type="checkbox"/>	Clear History
<input type="checkbox"/>	Clear All
<input type="checkbox"/>	Clear All Hidden
<input type="checkbox"/>	Toggle Overlapping

## Operators

some test commands

<input type="checkbox"/>	Test Mutation without Recomputation
<input type="checkbox"/>	Test Mutation with Recomputation
<input type="checkbox"/>	Dynamic Seeding
<input type="checkbox"/>	Make generation out of all history records