Probes

Introduction to OpenSim Probes

Probes are model components that perform vector measurements on an OpenSim model during a simulation. These measurements can then be output to an external file using a ProbeReporter analysis. Probes include:

- SystemEnergyProbe: measures the total energy state of a model.
- Umberger2010MuscleMetabolicsProbe: measures muscle metabolic power (based on Umberger et al., (2003 and 2010))
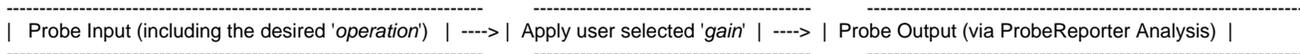- Bhargava2004MuscleMetabolicsProbe: measures muscle metabolic power (based on Bhargava et al., (2004))

OpenSim also includes additional probes to measure the force and power of an OpenSim Actuator and Joint internal power; however with the addition of OpenSim Outputs as of OpenSim 4.0, we recommend using the Outputs and OutputReporter instead of the Probes. See Analyses.

In addition to querying a model value, Probes can apply an *operation* to that value. Available operations include:

- "*value*": No operation performed: this just returns the probe input value.
- "*integrate*": Returns the integral of the probe value. Note that you must specify initial conditions.
- "*differentiate*": Returns the derivative of the probe value.

Finally, a scaling factor may also be applied to the output, after the operation has been applied. If a ProbeReporter analysis is attached to the model during a simulation, this will cycle through all Probes in the model and report them to a single file.

The data flow for a probe can be visually represented as follows:

```
----------------------------------------------------------------     ----------------------------------------     ----------------------------------------------------------------
|  Probe Input (including the desired 'operation')   | ---->|  Apply user selected 'gain'  |  ---->  |  Probe Output (via ProbeReporter Analysis)  |
----------------------------------------------------------------     ----------------------------------------     ----------------------------------------------------------------
```

Specifying Probes in the OpenSim Model

Adding and Deleting Probes in the GUI
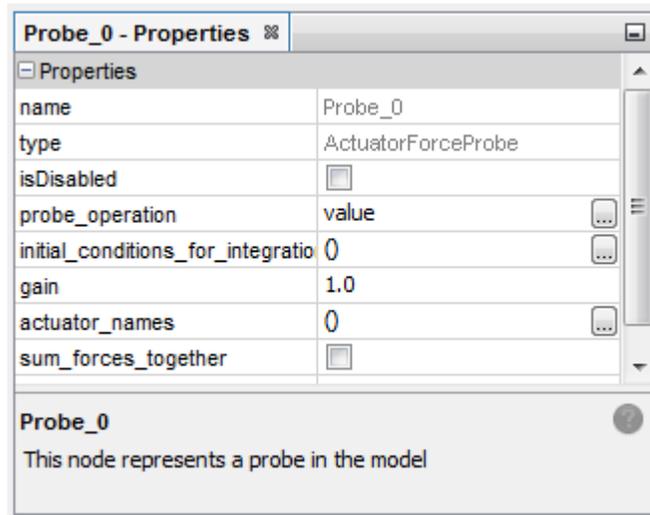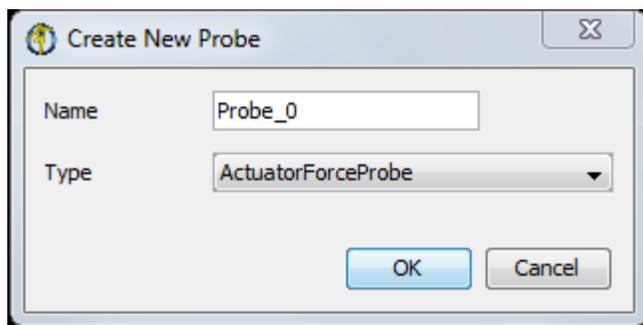
You can add and delete probes in the OpenSim GUI.

To add a probe:

1. Go to the Model of interest in the **Navigator** and find the **Probes** node.
2. Right click and select **New Probe...**
3. Give the probe a name and choose the type of probe (e.g. ActuatorForceProbe).

Specify the probe's parameters:

1. Find the probe you've added and bring up its properties in the **Property Editor**
2. You can use the Property Editor to specify the relevant parameters.
3. Note that for some properties (e.g. MetabolicParameterSet), you will need to make changes in the XML file directly

See the section below for more information about the parameters of a probe.

To delete a probe, Right click on the probe of interest in the Navigator panel and select Delete.

ⓘ **Using probes with the gait workflow?** Please note that the Residual Reduction Tool makes a copy of your model and then removes all the actuators. If you've added probes to the model that rely on any of these actuators, OpenSim will disable them. If you run RRA and choose to Adjust the model, you'll need to re-enable your probes.

Probe Parameters

Probes are specified within a <ProbeSet> in the *.osim model file. Although individual probes may have their own properties, all probes should contain the following properties:

- *isDisabled*: specify whether or not the probe is disabled or enabled.
- *operation_to_apply_to_probe*: a string that contains the operation to be performed on the probe value (available operations: 'value', 'integrate', 'differentiate').
- *initial_conditions_for_integration*: initial conditions for the integration. Only required if the "*integrate*" operation is selected.
- *gain*: gain to apply to the result, post-operation.
- *sum_forces_together*: specify whether to report the sum of all actuator forces or report the sum and each force value separately

To get a list of required properties for a specific probe, use the XML browser or Property Editor in the GUI, type "analyze –PI <ProbeName>" from the command prompt or see the Doxygen for Probes.ProbeReporter Analysis (for reporting probe outputs to file)

When Probes are specified in an OpenSim model, they will be automatically evaluated during a simulation. However the probe output values will not be reported to file unless a ProbeReporter analysis is also attached to the simulation (more information on OpenSim Analyses). The ProbeReporter 's job is to cycle through all 'enabled' Probes in the model at a user specified simulation step interval, and report their values to an external file. Note that an Analysis is *not* a ModelComponent, and therefore *not* specified as part of the model. Rather, OpenSim analyses are set up as part of the tool that is invoked to perform a simulation. E.g. inside the RRA/CMC/Forward setup file under <AnalysisSet>.

ⓘ For probes set to "integrate" you must attached the ProbeReporter Analysis to the Forward Tool or CMC Tool. The Analyze tool does not perform an integration.

ProbeReporter

```
<AnalysisSet name="MyAnalyses">
        <objects>
                <ProbeReporter name="MyProbeReporter">
                        <!--Flag (true or false) specifying whether whether on. True by default.-->
                        <on> true </on>
                        <!--Specifies how often to store results during a simulation. More
                                specifically, the interval (a positive integer) specifies how many
                                successful integration steps should be taken before results are
                                recorded again.-->
                        <step_interval> 10 </step_interval>
                </ProbeReporter>
        </objects>
        <groups/>
</AnalysisSet>
```

Creating your own probe (for advanced OpenSim API users)

Custom child class Probes may be created by overriding the following pure virtual methods:

- SimTK::Vector computeProbeInputs(SimTK::State& s) : defines the vector value of the probe as a function of the current system state s.
- int getNumProbeInputs() : defines the number of probe inputs in the return vector from computeProbeInputs().
- Array<std:string> getProbeOutputLabels() : defines the labels for the probe outputs (used in the ProbeReporter Analysis for reporting to file).

See the Probe Doxygen for more details on using the API to create custom Probes.

Next: OpenSim Models

Previous: Joint Reactions Analysis