SYNCOPATION SOFTWARE

TECH NOTE

Reducing Runtimes in DPL

DPL has the most powerful decision tree engine on the market today, but even so, modeling complex decision problems or problems with particular structures may result in long run times. The number of endpoints (also called leaf nodes) in a decision tree grows exponentially with the number of nodes. A decision tree with 170 nodes, each with three states, has approximately as many endpoints as there are atoms in the universe. While it's unusual for a model to have more than a few dozen distinct and important sources of uncertainty, if some chance nodes must be repeated in several time periods (e.g., in a real options learning model), or if they apply independently to several projects in a portfolio, the number of endpoints can quickly mount.

This page lists several techniques which will speed up the execution of some models. These range from changing settings in a dialog box to making significant structural changes. Contact technical support if you have any questions. Before putting any effort into speeding up a model, it's worth thinking about whether it can be pruned-- are there nodes that have little effect (bottom of the tornado diagram) or spreadsheet calculations which are needlessly detailed?

1) Convert the spreadsheet to DPL code.

If your analysis includes a spreadsheet, Excel is probably consuming far more processor time than DPL (tip: you can see this in the Windows Task Manager). When your Excel model has been translated into DPL's own language and compiled, DPL can evaluate it intelligently, caching frequently used formulas and only recalculating sections that change between scenarios. See the section on Spreadsheet Conversion in the User Manual for more information.

2) Reduce the number of levels gathered for decision policy.

By default, DPL saves all of the "endpoint data" when you run a decision analysis. This means you can click all the way down to the endpoints in the Policy Tree^{TM}. For models with millions of endpoints, this may be more than you want to know. You can speed up the run (and reduce the amount of memory required) by decreasing the number of decision policy levels in the Home | Run group.

3) Reduce the number of intervals for output distributions.

By default, DPL groups endpoint data (probabilities and outcome values) into 500 intervals when forming the risk profile. This allows DPL to produce a risk profile of reasonable size even for models with millions of endpoints. The more intervals DPL uses the smoother the risk profile will be. However, if your model has a large number of endpoints, and you're running it a number of times, you may not need a smooth risk profile from each run. Click Home | Run | Options to change the number of output distribution intervals. If you don't need a risk profile at all, deselecting it in the Home | Run group will further reduce runtime.

4) Reduce the number of levels of status display.

If you like to have DPL show its progress on the decision tree during a run, make sure you don't have too many levels displayed. If the branches are fluttering faster than you can see them, DPL may be spending a significant amount of processor time updating the screen. A few levels of status display at the top of the tree will have a negligible effect on runtimes. Similarly, watching the Excel spreadsheet recalculate can slow things down considerably, so it's best to keep Excel in the background (or minimized) during the analysis.

5) Experiment with the various optimizations.

Most of DPL's optimizations are turned on by default. If an optimization is not effective for a particular model, it may slow things down. Try running your model with and without the various optimizations in the Home | Run | Options dialog. For a particularly tough model, Full Tree Enumeration may be faster than Fast Sequence Evaluation (Home | Run | Decision Analysis drop down menu).

6) Use Monte Carlo simulation.

If you don't require an exact expected value, you can use Monte Carlo simulation with a moderate number of samples to obtain approximate results. Simulation is best for models with a large number of chance nodes and few or no "downstream" decisions. It's less effective for real options type models with decisions in several time periods.

7) Place computationally-intensive chance nodes higher up in the tree.

Chance nodes in the decision tree can be reordered without affecting results as long as they are not moved across decisions. Runtimes can be reduced by placing the most computationally intensive nodes at the top of the tree, so they won't change frequently as DPL moves through the scenarios. Note that computationally intensive nodes may or may not be value sensitive. For example, the discount rate may be highly value sensitive, but it is unlikely to be computationally intensive, because only a few NPV's need to be recalculated when the discount rate changes.

8) Use multiple get/pay expressions.

DPL has more opportunities to optimize a model with multiple get/pay expressions than one with only a single get/pay at the end of the tree. Place get/pay expressions as high in the tree as possible.

9) Avoid multiple attributes.

Models are often set up with multiple attributes to produce Time Series Percentiles. Keeping track of endpoint data for multiple attributes takes time, so if a particular run does not require all the attributes, you may want to temporarily remove them (duplicate the model in the Workspace Manager).