

Synopsis of HPC Excel 2010 Insurance Models Achieving Significant Performance and Precision Improvements

Conversion of Dynamo to HPC

“Dynamo” is a Dynamic Financial Analysis (DFA) model placed in the public domain by Pinnacle Actuarial Resources. A version of this model also is available from the Casualty Actuarial Society (CAS). The model consists predominantly of spreadsheet cellular logic. Visual Basic for Applications (VBA) code in this workbook is minimal and use primarily for a serial processing of simulations. The cellular logic has dependencies on about 750 inverse probability distributions valuations. These dependent cells both simulate cash flows and the accounting of those cash flows over a 5-year time frame.

Repeated, embarrassing parallel looping across simulations produces probability distributions for a wide variety of financial statement metrics such as policyholders surplus. There is an accounting overlay on simulated cash flows within dynamic financial models. Parallelization of Dynamo enabled it to run ten times faster in a high performance computing (HPC) cluster. Originally, Dynamo was designed for 1,000 simulations. It does 750,000 simulations for much improved precision of risk metrics in about the same time as it did standalone. This work illustrates the comparative ease with which performance of many actuarial models can be greatly enhanced using HPC. ***The HPC parallelization task was completed within several days, and the resulting VBA code required for HPC Excel 2010 SOA operation is extensible to any simulation workbook embodying similar characteristics of cellular logic to determine a simulation trial's results.*** This type of HPC parallelization is an “F9er” because each simulation is achieved essentially by virtual pressing of the F9 calculate key.

Stochastic Simulation of Provisions for Block of Equity-Linked Life Insurance Policies

This parallelization task for this life model is referred to as a “Seriatim” type of HPC parallelization because the modeling is heavily dependent on VBA code for calculations (rather than cellular logic) and there are many policies to analyze for the seriatim. A 50,000 block of life insurance policies is analyzed for each simulated interest rate path. (I.e., the object of parallelization was not policies; rather, it was the financial scenario that all policies could simultaneously see as a realization of investment results.) The cash surrender values of the policies are dependent on simulated investment returns. The approach follows principles-based reserving and uses stochastic rendering of probabilistic cash flows and their valuation in a (modeled) market-consistent fashion. Termination either by death or lapsation was simulated using mortality and lapsation probabilities. Each compute node on the HPC cluster evaluated all policies for an investment path. The analysis was similar to the derivation of provisions under Solvency II.

This serial valuation involving direct simulation of policy cash flows is so computationally demanding, it would take a single computer 1.5 months. On a 220-core HPC cluster, the HPC Excel 2010 SOA job was completed in less than nine hours. ***This model illustrates the usefulness of HPC for brute-force simulations and analyses that otherwise are prohibitively slow to do.***

(Re)insurance Property and Casualty Pricing Model Using a Parametric Sweep

This model also is a Serialim-styled workbook; however, each partition stratum involves a single combination of probability distribution parameters and censors for both the frequency and severity distributions. The HPC version is similar to the pricing of policies falling into different classifications or strata. A parametric sweep involves the specification of a minimum, maximum and change interval between those points. The sweep moves from the minimum to the maximum in intervals. Many combinations of interval points are obtained among all of the many censors (deductibles, limits of liability, aggregate limits of liabilities and contagion) and parameters for a Poisson frequency distribution and truncated normal severity distribution. For any number of combinations, the performance on an HPC cluster easily exceeded ten times faster than a standalone calculation. However, the aggregate loss distributions simulated with high-volume simulation (>500,000 trials) produces markedly better precision so performance enhancement is not the only objective of using HPC. Further, ***this is an example of how HPC can be used to price an entire set of classifications (a combination of parameters and censors).*** Each compute node on the cluster analyzed a single combination of parameters and censors to produce an aggregate loss distribution for that combination.