

Select Font Size: [A](#) [A](#) [A](#)Sponsored By  
**SPECTRUM**

## PlayStation 3 Processor Speeds Financial-Risk Calculation

By Samuel K. Moore



PHOTO: DON FARRALL/GETTY IMAGES

19 November 2008—Of the many things that have gone wrong on Wall Street this past year, the use and misuse of computational algorithms meant to give financiers a clear picture of the risk of big losses was one of them. One important calculation, called Value-at-Risk (VaR), is a way of assessing the probability that an investment portfolio will lose a specified value over a certain period of time. Though VaR's reputation is much maligned, experts say firms have little choice but to continue, if not accelerate, their use of computational algorithms as the need to calculate risk and value has become more acute. The surviving financial firms might get a little help from code produced for [Cell](#), the processor behind [Sony's PlayStation 3](#) as well as a number of high-end IBM servers and supercomputers.

By taking advantage of Cell's unique architecture, computer scientists at Georgia Tech, in Atlanta, say they have found a way to accelerate the generation of random numbers up to 33-fold compared with what's possible using commercially available Intel or AMD processors. Random-number generation is key to many so-called Monte Carlo simulations, but as a proof of concept, the researchers built their [random-number generator](#) into a program that efficiently runs a VaR algorithm. David A. Bader, executive director of high-performance computing at Georgia Tech, says his group has been working with several financial firms, whose names he would not disclose due to legal agreements, on their use of the VaR software. Bader says the source code behind the random-number generators for Cell will be made available during the SC08 supercomputing conference this week in Austin, Texas.

"What we're able to do is generate not just a single random number fast, but streams of random numbers that can be fed into Monte Carlo simulations," says Bader, who is also a professor of computational science and engineering.

The basis of many scientific and financial computational algorithms, Monte Carlo methods all start with some random numbers, then perform an operation on them, and repeat the process to arrive at a result. (Imagine playing the game Battleship, for example.) Often the most time-consuming part of a Monte Carlo computation is the generation of a good set of random numbers (or pseudorandom numbers, as computers don't produce true random numbers). Bader and his graduate students Aparna Chandramowlishwaran and Virat Agarwal adapted two well-known random-number algorithms to Cell: the 64-bit linear congruential generator (LCG) and the 32-bit Mersenne twister. With the LCG, their speed increased 33-fold and with the Mersenne twister, between 14-fold and 29-fold.

The key to accelerating both algorithms lies in [Cell's architecture](#), says Bader. Cell comprises nine processor cores. One, the Power Processing Unit, is a general-purpose processor that can run an operating system and do other control tasks, such as coordinating the work done by the other eight processors. Those eight—called the Synergistic Processing Units (SPUs)—are designed to

efficiently perform tasks common to multimedia applications: video compression and decompression, encryption and decryption, rendering and modifying graphics, and the like. The Georgia Tech innovation was in finding a way to parallelize the LCG and Mersenne twister algorithms in a way that played to the SPUs' strengths. For both algorithms, that meant adapting the core algorithm to run on all eight SPUs but initializing each of the processor cores with a different set of parameters in order to guarantee that their outputs, when combined into one stream of numbers, would not have repeating or similar sections.

Though it can be a rate-limiting step in other Monte Carlo calculations, generating a good set of random numbers is far from being the main drag on VaR, says Dan Rosen, financial engineer and president of R2 Financial Technologies. "Really, it's what you do with those random numbers," says the Toronto-based algorithms expert. Most of the computational work involves calculating the values of stocks and other instruments in each of hundreds or thousands of different scenarios needed to produce an accurate understanding of risk. Consequently, most of the algorithmic research in VaR is dedicated to speeding the valuation process or reducing the number of scenarios needed to get to a given level of accuracy, says Rosen.

Even so, it seems that financial firms are interested in what Bader's group has done. "We are working with several of the major financial institutions that do this type of calculation," says Bader. Despite their troubles, "financial institutions aren't going to stop all calculation," he says. "In fact, they're doing quite the reverse—increasing the big calculations they're doing to better understand pricing."

Still, the ability to model risk faster is of little use if the model doesn't reflect reality. "While speed on a technical level might be valuable, it's not going to substitute for what should be included in the model," says Robert Hoyt, professor of risk management at the University of Georgia, in Athens. "I think what's going on will infuse a dose of reality—that models always have limitations, and in some cases, serious limitations."