White Paper

High-Performance Risk Analysis with REvolution R

Abstract

We illustrate the substantial performance gains possible with REvolution R Enterprise and the Intel® Xeon® processor 5500 series when computing risk metrics with the CreditMetrics[4] algorithm.

REvolution R Enterprise

REvolution R Enterprise is a commercially-supported distribution of the open source R language from REvolution Computing. The R language is a very popular open-source data analysis and statistics language with a large and rapidly-growing following in the quantitative and risk analysis communities.

REvolution Computing provides commercially-supported, highly tuned and optimized distributions of the R language for a variety of operating systems and architectures.

R is a powerful and efficient functional programming language with a natural and easy to understand syntax. R includes a rich set of first-class time-series objects, making it particularly well-suited to financial applications. R's origins as a data analysis language make it very easy to access and operate on a wide variety of financial data. The open-source community (including many contributors from industry) has contributed a large number of packages for financial computation with R. Major contributions include the Performance Analytics[1], Rmetrics[5], and Quantmod[3] packages. Contributed packages include hundreds of programs and functions for financial engineering, risk modeling and econometrics. This whitepaper focuses on performance benchmarking of the CreditMetrics package by Andreas Wittmann.

Ever-increasing data volumes and model complexity often force practitioners to limit their use of high-level languages like R to the prototyping phase, relying on lower level languages for production work. However, models built in C, C++, Fortran, Java and the like are often slow to deploy, costly to produce, and typically much more error-prone than algorithms developed in high-level environments like R.

REvolution R Enterprise, built with the Intel® Math Kernel Library (Intel® MKL)[7], provides optimized algorithms and tools for high-performance computing with R in enterprise environments. REvolution R reduces the need to rely on custom low-level programming in many circumstances, improving productivity and accelerating code deployment from prototyping to production.

Intel® Xeon® Processor 5500 Series

An Overview of Intel's New Microarchitecture

Intel Core microarchitecture combines several innovative features to deliver intelligent and scalable performance that is significantly more energy efficient than previous architectures.

Dynamic scalability—Performance when you need
Dynamic management of cores, threads, cache, interfaces, and power delivers energy-efficient performance on demand.

Intel® Turbo Boost Technology—Automatic performance improvements
Automatic increases in processor frequency by up to 400 MHz can increase application performance, especially for single-threaded workloads.

Intel® Hyper-Threading Technology—Maximum parallel application performance
Running multiple threads on each processing core—up to 16 threads in dual-socket quad-core systems—can increase total application performance while requiring only a fraction of the power that would be necessary to support additional cores.

Integrated Memory Controller—Direct access to performance
Each processor accesses three channels of directly connected, native DDR3 memory at speeds up to 1333 MHz through an Integrated Memory Controller, achieving up to 18.11 GB/sec of memory bandwidth per socket.

Intel® QuickPath Architecture—Applications unleashed
High-speed, direct connections among microprocessors and the I/O hub provide each microprocessor with fast, coherent access to the memory attached to the other microprocessor. Together, the Integrated Memory Controller and Intel QuickPath Architecture deliver more than three times the memory bandwidth per node compared with the previous-generation dual-socket architecture, enabling you to unleash memory-bound applications[6].

Shared cache—Reduced data latency
An inclusive, fully shared L3 cache facilitates rapid access to frequently used data and smoother scaling of cache performance as you add cores to your HPC cluster.

More instructions per clock cycle—Greater raw, energy efficient performance
Support for greater parallelism, more efficient system algorithms, and enhanced branch predictions deliver more instructions per clock cycle, helping you to accelerate innovation and discovery while conserving energy.

By improving both computational resources and aggregate system bandwidth, the new microarchitecture helps to create balanced HPC systems that can produce a dramatic increase in total application performance while also increasing cluster density and energy efficiency. Achieve greater performance per meter and per watt than ever before.

CreditMetrics

CreditMetrics is a methodology for estimating credit value-at-risk (CVaR). Published by J. P. Morgan in 1997, CreditMetrics has become an industry benchmark for evaluating credit risk across portfolios consisting of many kinds of credit instruments.

The CreditMetrics algorithm for evaluating portfolio risk is a quantitative model of volatility in the value of each instrument in a portfolio due to changes in credit rating or default. The model also takes in to account correlations between events to provide an aggregate measure of volatility. A schematic diagram of the CreditMetrics framework appears in Figure 1. The CreditMetrics CVaR model constructs models which help explain volatility in credit-related instruments.

The CVaR model for large portfolios involving more than a few hundred instruments can be expensive to compute. The most computationally demanding part of the CVaR model involves generating

correlated, normally distributed random variables. A number of methods exist for computing correlated random variables including the Singular Value Decomposition and Cholesky factorization e.g., Leon-Garcia[2]. The CreditMetrics R package [4] uses the Cholesky factorization of a (symmetric positive definite) correlation matrix.

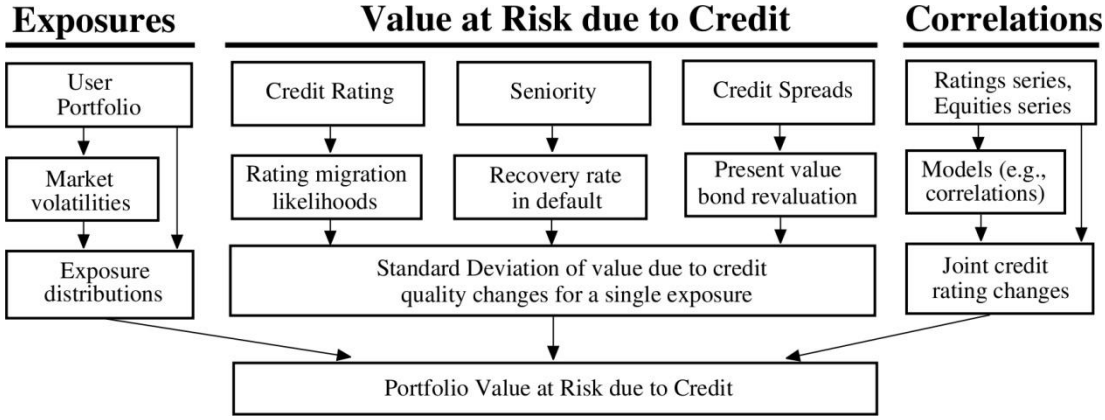


Figure 1: CreditMetrics schematic (from the CreditMetrics–Technical Document, Copyright c 1997, J. P. Morgan & Co. Inc.).

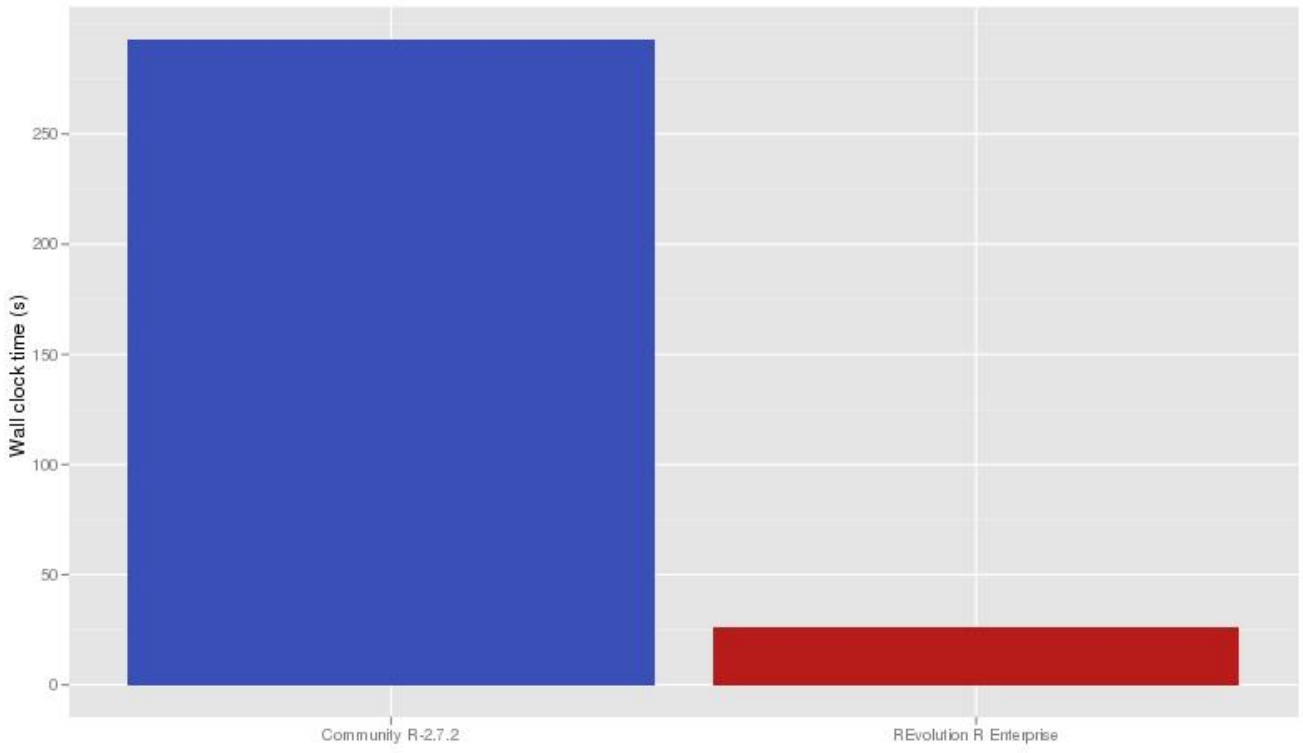
REvolution R Enterprise includes optimized numeric routines for a wide variety of numerical methods including cutting-edge fast and efficient Singular Value and Cholesky matrix decompositions. The routines are multi-threaded to exploit thread-level parallelism on modern multi-core processors. Even on single-core systems, REvolution R routines greatly outperform reference routines by exploiting processor cache and vector instructions when appropriate. Because most REvolution R optimizations occur at a low level, many high-level operations benefit without any program-level modification.

Portfolio Performance Benchmarks

We compare the performance of a reference download of R-2.7.2 from http://cran.r-project.org/ with the REvolution R Enterprise distribution, based on R-2.7.2. Version 0.0-2 of the CreditMetrics R package (also available on CRAN) was used in each test. The test benchmark used a synthetic portfolio of 2,000 instruments and 50,000 simulations in the CVaR computation. We tested on two test platforms: an Intel Xeon processor 5400 series with dual quad-core Intel Xeon X5482 processors running at 3.2GHz and 16 GB RAM, and an Intel Xeon processor 5500 series with dual quad-core Intel Xeon X5560 2.8 GHz processors and 16GB RAM. Each system was equipped with the Red Hat Enterprise Linux 5.3 GNU/Linux operating system.

The bulk of the computational work involved in the CVaR computation involves the production of correlated random numbers used in the Monte Carlo portion of the code. Figure 2 shows a comparison of R-2.7.2 and REvolution R Enterprise for the CreditMetrics CVaR computation for a portfolio of 2,000 instruments. Even in the presence of essentially sequential pre- and post-processing code around the core computation, REvolution R Enterprise exhibits significant performance gains--more than 8 times faster than R-2.7.2. Even with a 12% slower clock rate, the Intel Xeon processor 5500 series outperforms the previous-generation Intel Xeon processor 5400 series by 27% when using REvolution R Enterprise. Interestingly, the new Intel Xeon processor 5500 series architecture significantly outperforms the Intel Xeon processor 5400 series even on un-optimized code.

*Figure 2: Comparison of R-2.7.2 and REvolution R Enterprise CVaR computation on a portfolio of 2,000 instruments (*Intel Xeon processor 5500 series, lower is better*).*



Summary

REvolution R Enterprise can substantially reduce computation time for the benchmark CreditMetrics analysis on multi-core workstations. The optimized numeric routines available in REvolution R Enterprise can transparently speed up a large number of compute-intensive tasks in R. High-performance numerics, commercial support, training and developer tools for R from REvolution Computing position the R language for rapid prototyping and production environments in quantitative finance.

	Intel® Xeon® Processor 5400 Series	**Intel® Xeon® Processor 5500 Series**
Community R-2.7.2	588	293
REvolution R Enterprise	33	26

Table 1: Wall-clock time (s) CVaR Benchmark (lower is better)

References

[1] Peter Carl and Brian G. Peterson, Performance Analytics R Package, http://braverock.com/R/.
[2] Alberto Leon-Garcia, Probability, Statistics and Random Processes for Electrical Engineering, Prentice Hall, 2007, p. 631.
[3] Jeffrey A. Ryan, http://www.quantmod.com/.
[4] Andreas Wittmann (R Package),CreditMetrics Technical Document, http://www.riskmetrics.com/publications/techdocs/cmtdovv.html, J. P. Morgan/RiskMetrics Group, April, 1997.
[5] Diethelm Wuertz, http://www.rmetrics.org/, ETH Zuerich.
[6] Intel internal measurement (Feb. 2009). STREAM-Triad benchmark. Red Hat Enterprise Linux* Server 5.3. Intel® Xeon® processor E5472, 3.0 GHz, 2x6MB L2 cache, 1600MHz system bus, 16GB memory (8x2GB FB DDR2-800) vs. Intel Xeon processor X5570, 2.93 GHz, 8MB L3 cache, 6.4QPI, 24GB memory (6x4GB DDR3-1333).
[7] Intel® Math Kernel Library, http://software.intel.com/en-us/intel-mkl/