



Highly Productive, Scalable Actuarial Modeling

Delivering High Performance for Insurance Computations Using

Milliman's MG-ALFA[®]

Microsoft[®] Windows[®] HPC Server 2008

IBM[®] BladeCenter[®] and System x[™] Clusters

IBM Computing on Demand



Microsoft[®]



Table of Contents

Table of Contents	1
Introduction	1
Windows HPC Server 2008: A Seamless Cluster Computing Solution	2
IBM BladeCenter and System x Clusters	3
IBM Computing On Demand: Scalable Dynamic HPC Infrastructure	4
How it works	4
An Integrated, Scalable, Dynamic HPC Solution for MG-ALFA	5
Performance	7
Cluster Configuration.....	7
Performance Tests	7
Performance Measurements	8
Test Results and Tips for Optimizing Performance.....	9
The Bottom Line	10
APPENDIX A: MG-ALFA Test Drive via IBM Computing on Demand	14
APPENDIX B: Additional Information	18

Introduction

Throughout the entire financial services industry, there is increasing demand for rapid and accurate analyses of the risk associated with investments and financial positions. In the life insurance sector, in particular, the combination of new and pending regulatory requirements and the recent market volatility are forcing companies to create highly detailed models that must be simulated using thousands of scenarios. With the rapidity of changes in the market, such simulations must be performed far more frequently than in the past, leading to a huge demand for computing power that can only be met through the use of dynamic cluster-based high-performance computing (HPC) infrastructures capable of providing the data access and processing power necessary to run large simulations over extended periods of time.

Milliman, Inc., one of the world's largest actuarial and consulting firms, is the developer of the MG-ALFA[®] (Asset Liability Financial Analysis) software application that is widely used to carry out detailed financial projections in support of product development, financial reporting, risk management, and decision analysis. MG-ALFA is used by or on behalf of insurance companies, governments at all levels, rating agencies, and many other organizations to analyze insurance portfolios, pensions and benefits, and other complex financial instruments. Because MG-ALFA is an interactive Windows[®]-based desktop application with a well-developed graphical user interface, users are able to build complex models quickly and easily—the only difficulty arises when it comes time to simulate the models, because the large number of calculations can easily swamp even the fastest of today's desktop computers.

Now, however, high performance computing based on Microsoft[®] Windows[®] HPC Server 2008 (HPCS) can provide the horsepower required for MG-ALFA users to run extremely complex simulation models. For example, a 30-year simulation of a 1,000-scenario model might well require hundreds of millions of cash flow projections. Even on a fast desktop, such a calculation could easily take several hundred hours, making it nearly impossible to incorporate the results into a timely decision-making process. On the other hand, even a modest-size HPCS cluster using modern multi-core processors can reduce the time required to just a few hours or even minutes, entirely changing the way that the model might be employed in the ordinary course of business. And because the HPCS cluster provides an operating environment that is based on the same operating system technology as Windows desktops, the transition from desktop computation to HPC cluster computation is completely seamless and nearly invisible to the end user.

This white paper presents benchmark results obtained running MG-ALFA on an IBM[®] System x[™] iDataPlex[™] dx360 M2 cluster with quad-core Intel[®] Xeon[®] x5550 (Nehalem) processors running Windows HPCS. ***The benchmark results are outstanding, demonstrating not only raw computational speed, but also excellent scalability using up to 200 computational cores.***

To deliver such outstanding performance for real problems on a day-to-day basis, modern HPC solutions depend on three major interlocking components: fast computational hardware infrastructure, a software operating environment providing job and resource management and high-performance data access, and a set of tuned end-user applications. This white paper describes a cost-effective, flexible, scalable, and integrated solution for complex actuarial analyses created by bundling ***MG-ALFA with Microsoft Windows HPC Server 2008 on IBM[®] BladeCenter[®] or System x[™] clusters.*** This solution may be deployed in a traditional office / data center environment or via the IBM Computing on Demand (CoD) cloud computing service. ***This optimized actuarial platform delivers the computational power and energy efficiency required by the insurance industry by providing secure access to cluster resources capable of handling the compute-intensive workloads generated by MG-ALFA.***

Subsequent sections of this white paper address the following topics:

- Microsoft Windows HPC Server 2008;
- IBM BladeCenter and System x clusters;
- IBM Computing on Demand;

- Integrating MG-ALFA with Windows HPC Server 2008 and IBM Computing on Demand; and
- MG-ALFA performance on HPCS-enabled IBM System x clusters.

Two appendices provide information about how to test drive MG-ALFA at one of the IBM Computing on Demand centers, and where to find additional information about the companies, products, and technologies discussed in this white paper.

Windows HPC Server 2008: A Seamless Cluster Computing Solution

The preferred HPC platform for running MG-ALFA computations in parallel is Windows HPC Server 2008 (HPCS). Windows HPC Server 2008 combines the power of a 64-bit Windows Server® platform with rich, out-of-the-box functionality to improve the productivity, and reduce the complexity, of an HPC environment. It is an ideal fit to MG-ALFA because it dovetails seamlessly with the desktop Windows environments required by MG-ALFA.

Windows HPC Server 2008 is composed of a cluster of servers including a single head node and one or more compute nodes (see Figure 1). The head node, which may provide failover via Windows Server 2008 Enterprise high availability services and SQL Server clustering, controls and mediates all access to the cluster resources and is the single point of management, deployment, and job scheduling for the compute cluster. Windows HPC Server 2008 can use an existing Active Directory® service-based infrastructure for security, account management, and overall operations management using tools such as System Center Operations Manager.

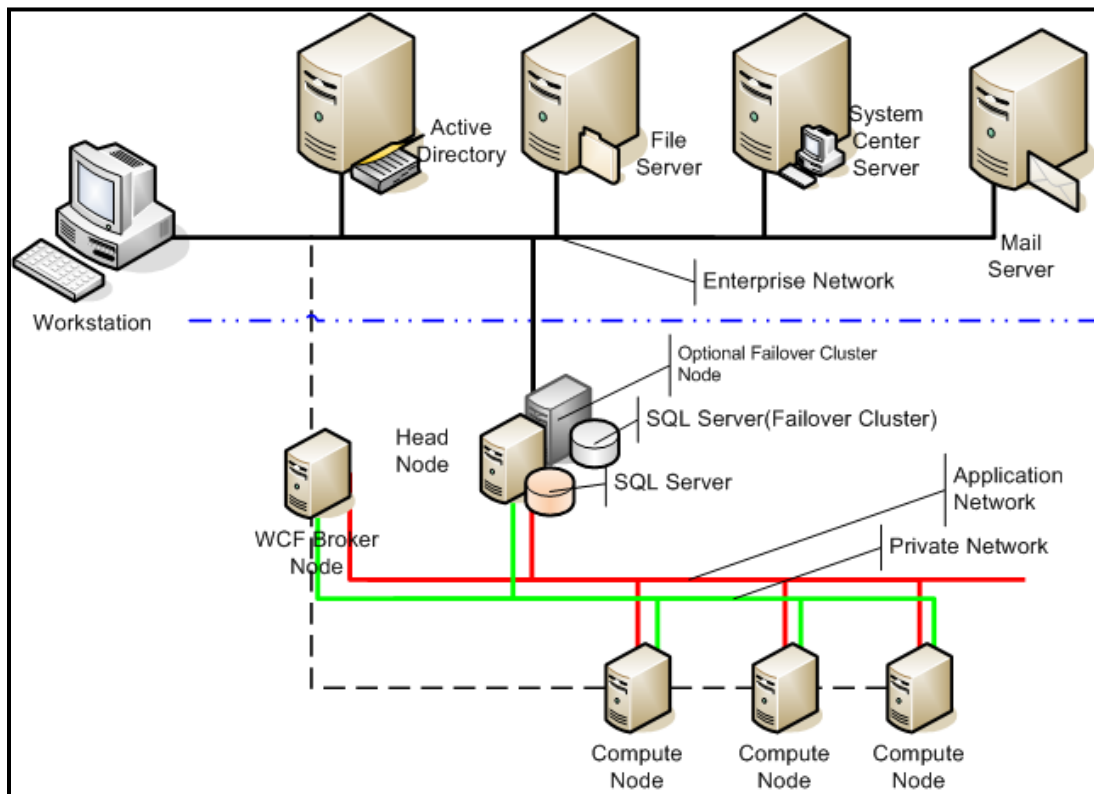


Figure 1: Illustration of HPCS Cluster Architecture (Source: Microsoft Corporation)

Windows HPC Server 2008 brings the power, performance, and scale of high performance computing (HPC) to mainstream computing by providing numerous end-user, administrator, and developer features and tools. Among these are:

- quick deployment using built-in wizards and management consoles; comprehensive management, administration, and diagnostic tools;
- flexible job scheduling and management;
- high-speed network interconnects based on NetworkDirect; high-performance storage such as IBM's General Parallel File System (GPFS) and iSCSI storage area networks (SANs);
- integrated application development environments like Microsoft Visual Studio that provide access to a variety of standard parallel programming environments (such as OpenMP, Message Passing Interface (MPI), and Web Services) and a parallel debugger; and
- overall ease of management derived from the integration of HPCS with the broad-based Microsoft ecosystem including Systems Center Operations Manager 2007, Windows Server 2008 and Microsoft Active Directory.

Together, MG-ALFA and HPCS form a complete, cost-effective software solution for high-performance actuarial modeling that provides insurance firms with greater agility and confidence in their decision-making while maximizing investments in HPC infrastructure and application software.

IBM BladeCenter and System x Clusters

For complex computational problems, Microsoft Windows HPC Server 2008 running on an IBM cluster can help accelerate time-to-insight by providing a high-performance computing platform that is energy-efficient and simple to deploy, operate, and integrate with existing infrastructure and tools. When combined with MG-ALFA, systems like this are an ideal solution for highly productive, scalable actuarial modeling.

For small and medium businesses, IBM offers the IBM BladeCenter S, providing the power of a data center in a desk-side form factor. The BladeCenter S is a true all-in-one solution: including servers, networking, management, and an optional fully redundant, integrated SAN storage system built into the chassis.

For larger installations, IBM offers the IBM System Cluster 1350 and IBM System x iDataPlex™. The IBM System Cluster 1350 capitalizes on IBM's extensive engineering, testing and deep clustering experience, utilizing cost-effective rack-optimized servers and BladeCenter high-density blade servers to offer extraordinary performance, flexibility, and reliability. The IBM System x iDataPlex™ solution revolutionizes data center economics by packing double the number of servers in a single side-by-side rack chassis, using significantly less energy, and providing simplified management in a modular design.

IBM clusters make use of a variety of compute and storage servers that are built on open standards and offer a range of affordable, high performance, easy to manage platforms designed to help optimize datacenters and lower total cost of ownership. Among these are the following:

- Intel®-based IBM System x3550 and x3650 servers offering strong unprecedented performance and reliability for the data center.
- IBM BladeCenter HS22 high-density blade servers providing an efficient, integrated solution based on two-socket Intel® Xeon® processors and up to 96 GB of internal memory.
- IBM BladeCenter HS21 XM high-density blades providing expanded memory and processing power to enterprise environments to deliver optimal performance with low-voltage processors in an energy efficient, high availability, integrated system.

- IBM System x iDataPlex servers, including the dx320, dx340, dx360, and dx360 M2 servers, designed to provide high performance, energy efficiency, and cost-effectiveness in a compact package.

The Intel-based IBM iDataPlex dx360 M2, System x3650 M2, and System x3550 M2 are some of the most power efficient servers in the market today. As of October 8, 2009, the iDataPlex dx360 M2 system was ranked the best power performance system, delivering an outstanding 2,231 Java ops/watt on the SPECpower_ssj2008 benchmark (See http://www.spec.org/power_ssj2008/results).

IBM Computing On Demand: Scalable Dynamic HPC Infrastructure

Taking full advantage of MG-ALFA's enhanced modeling techniques, including multi-dimensional stochastic modeling, requires a flexible, dynamic infrastructure—one that can provide the data and processing power necessary to run simulations over extended periods, with thousands of scenarios and tens of thousands of model points. In a traditional owned-capacity model, companies purchase and deploy just enough computing capacity to manage anticipated “average” day-to-day processing. With this model, when project timelines overlap and peak computing demand is generated, analysts may be stuck waiting for the in-house processors to work through all the assigned tasks.

IBM Computing on Demand (CoD) is an IBM offering that provides companies with flexible access to vast computing power capable of handling large workloads. CoD users have access to security-rich supercomputing environments that can be used like on-site hardware, but without the capital commitment, management, and maintenance costs. When computing demands exceed in-house capacity clients can easily shift the excess workload to an IBM CoD center and purchase additional processing capacity necessary to help meet demand. The hardware is hosted, maintained, and supported by IBM to deliver cost-effective capacity that helps free companies to focus on business operations.

With access to CoD's on-tap supercomputing capability, an insurance company might be able to maintain business critical processor-intensive tasks in-house while, for example, delegating urgent MG-ALFA computations to reserved CoD capacity. Instead of incurring large capital investments to buffer capacity from spikes in demand, CoD users can treat additional capacity as a value-driven operational expense. If users only have to pay for capacity when they need it, the undedicated capital may be recommitted to strategic business objectives. Ultimately, IBM Computing on Demand can help companies achieve the speed and agility to lead the market, gain greater flexibility, and reduce costs.

IBM Computing on Demand provides:

- *Scalable Peak Capacity:* Access to HPC infrastructure to extend limited in-house capacity and meet short-term needs;
- *Deployment-free Infrastructure:* On-demand, fully managed clusters that can be reserved and accessed rapidly without installation and setup delays;
- *Variable Cost Advantages:* Agile pay-for-use model that can transform long-term fixed costs into business-driven operational costs; and
- *Superior Risk Management:* “Pay only when needed” capacity to help control and optimize IT expenditures.

How it works

IBM currently operates seven global Computing on Demand Centers throughout the world. Combined, these centers offer over 10,000 HPC compute cluster cores, 50 Terabytes of storage, all running the latest operating systems and connected by the fastest interconnects. IBM CoD centers feature a variety of technologies including Intel Xeon, AMD Opteron processors, or IBM POWER processors. Depending on the processor, the CoD centers support Microsoft Windows, Microsoft Windows Compute Cluster

Server 2003, Windows HPC Server 2008, Linux[®], or IBM AIX 5L[™]. Interconnects offered include 100 Kilobit or Gigabit Ethernet, InfiniBand[®], and Myrinet[®].

Customers purchase annual base memberships to the IBM Computing on Demand center of their choice. Base membership includes a “home” management node in the IBM CoD center and a software VPN connection (hardware upgrade available). Customers then create a customized computational environment on the management node, including their selected operating system (configured as required), software stack, and licenses. Customers maintain root control of the compute and storage resources within their assigned environment. A robust, security-rich networking infrastructure, including remote access through VPN, is designed to keep customer data and applications highly available.

With the customized environment already in place, customers can quickly and easily reserve and add computational capacity. Capacity is available through highly flexible and cost-effective contract terms, either from IBM or as a full service offering from Milliman who is able to offer their MG-ALFA clients “infrastructure as a service” via the IBM CoD offering along with application support. Compute power is billed per-processor with discounts for larger capacity commitments or longer rental durations. Storage capacity is priced per-gigabyte per-week.

An Integrated, Scalable, Dynamic HPC Solution for MG-ALFA

MG-ALFA enables users to build complex stochastic and nested-stochastic models that produce highly accurate forecasts and projections. Stochastic modeling is a complex mathematical process that uses probability and random variables to forecast financial values and performance. When multiple stochastic models are used in a hierarchical fashion (one inside another), the process is known as nested stochastic modeling. For example, a portfolio model might require monthly projections of income statement and balance sheet information over a 30-year period, where the reserve and capital values at each time point are determined using a stochastic projection. If the model included 1,000 scenarios and 1,000 projection paths at each valuation point, more than 360 million cash flow projections would be required for each instrument or insurance policy in the portfolio—a huge amount of computation that could require weeks of computation when run without the benefit of HPC.

An important aspect of stochastic models is that many of the calculations are completely independent of one another. For example, each of the scenarios is independent, as is each of the liabilities (insurance policies) in the portfolio. Since these calculations are independent, it is easy to see how to make simultaneous use of multiple computers in order to reduce the elapsed time for the calculations. Until recently, however, parallel computing of this sort (often called “embarrassingly parallel” because it requires almost no interaction among the computers) has been difficult to use and has not fit seamlessly with Windows-based applications like MG-ALFA.

That has all changed now. Over the past several years, Microsoft has released HPC software systems and worked with key vendors like Milliman to provide an environment for high performance computing that can be completely integrated with the desktop environment most business users use every day. For applications like MG-ALFA, Microsoft’s Windows HPC Server 2008 provides a path to exploiting HPC that fits into users’ business environments without the need for substantial custom infrastructure development either by the software vendor or end user.

HPCS supports a standard cluster comprising a head node and a number of compute nodes. The head node provides user interface and workload management services for the cluster, including job scheduling, job and resource management, node management, and node deployment. For reliability and high availability, an optional failover head node is supported as well. The compute nodes provide computational resources for the cluster. Additional servers running on management and infrastructure nodes (often pre-existing and sometimes external to the cluster) provide services such as DNS, DHCP, Active Directory, file storage, highly available databases, and others. Figure 1 on page 2 illustrates one possible architecture for an HPCS cluster.

Milliman has developed an option in MG-ALFA designed to exploit HPCS clusters. The MG-ALFA application architecture comprises a number of different interacting sub-applications, including tools for model building, simulation, and report generation. The simulation portion of MG-ALFA, of course, is the compute-intensive sub-application that can take advantage of high performance computing. It does this by exploiting the large amount of independence in the mathematics and implementation of complex models.

In its usual standalone desktop mode, MG-ALFA creates a number of run-specific input files that describe the computational process that is to take place. It places these newly-created files in a project work directory along with other model input files, files that tell its report writer what to run, and all required database executables. One of the newly-created files (Model.CMD) details how the project should be run—i.e., the name of the project, the database, the factor files that are to be used, variables to be captured, run mode, number of cycles to project, etc. The desktop instance of MG-ALFA then invokes a program (the “Bridge DLL”) that acts as a bridge to MG-ALFA’s calculation engine. The Bridge DLL uses the data in the Model.CMD file to make calls into the calculation engine to run the projection(s). When all calculations are finished, control returns to the desktop instance, which moves the results back to the project directory, properly updates any created factor files, and creates a table of contents for the reports.

In standalone mode, the calculation engine runs on the same machine as the desktop instance of MG-ALFA. The engine processes the large number of independent computational tasks (corresponding to scenarios or cells, for example) in a serial fashion, one after another. Since the tasks are independent, however, they could just as well be processed simultaneously by multiple copies of the calculation engine running in parallel on separate compute cores or machines. That is exactly what happens when an HPC backend server (a cluster, for example, where each compute node may have multiple CPUs or cores) is involved. Many copies of the calculation engine run on the backend server, and necessary data are routed to and from each of the engines to enable correct calculation of all the computational tasks.

This style of parallel computing is sometimes called “grid computing.” It is important to note that the pre-processing of the model is essentially independent of the details of the backend such as the type of processors, the number of engines, etc. Each task created by the desktop instance is simply a standalone serial computation that can be processed by any MG-ALFA calculation engine running anywhere.

More concretely, MG-ALFA interfaces to an HPCS cluster by carrying out the following steps:

1. The MG-ALFA desktop instance creates all of the runtime data files required for the entire computation and divides them into two groups: those that apply to the entire job, and those that apply only to an individual computational task. The job files are gathered into a compressed archive that is processed exactly once by each engine. The task files are also compressed but retrieved just by the engine that actually processes the task. All the files are placed in a shared folder accessible from each of the cluster nodes.
2. Next, the MG-ALFA desktop instance creates and submits to the Job Manager an HPCS job specifying the target number of engines to use and containing one HPCS task for each computational task.
3. When the job runs, the Job Manager allocates cores to the job and runs one MG-ALFA calculation engine on each core. The first engine that runs on a given node downloads and installs the archive of job files in a local folder on the node. Subsequently, engines running on the node need only download the task-specific files, generally a much smaller amount of data.
4. As the job proceeds, the Job Manager assigns individual computational tasks to each core. To run a task, the engine on the core accesses the data for that task, runs the calculation, and returns the output to a shared folder accessible from the desktop. The Bridge DLL on the desktop then performs post-processing on the task results as required. Meanwhile, the engine running on the back-end is assigned another task by the Job Manager.
5. When all tasks are complete, the Bridge DLL submits a “clean up” job to remove temporary data and files from the compute nodes.

While much of the creation and processing of the tasks is completely automatic, there are several ways that a user can impact the process and the achieved performance:

1. The user specifies the basis on which the tasks are created. For stochastic models, tasks are typically defined as one or more scenarios. For models running one or a few deterministic scenarios, tasks are defined by grouping liability cells (insurance policies or representative policies).
2. The user determines the size of each computational task by telling MG-ALFA how many scenarios or cells to include in each task. This is known as a “granularity” control, and setting it properly may have a significant impact on the overall computational efficiency.
3. The user specifies the target number of MG-ALFA engines to run on the HPCS server. This is actually the maximum number of engines; a smaller number may be used if the target number is larger than the number of tasks or the number of cores available to the job. (As jobs near completion, MG-ALFA tries to reduce the number of cores in use when the number of remaining tasks is less than the number of cores allocated by the Job Manager.)
4. The user specifies whether or not to copy the large job-related files to each node as part of job initialization. The alternative—having each engine read the files from a shared location such as a file server—works well in most cases.

Performance

This section provides information about MG-ALFA performance based on tests run on an IBM System x iDataPlex cluster running Microsoft Windows HPC Server 2008 at Microsoft’s Enterprise Engineering Center in Redmond, Washington. The tests were run using MG-ALFA Version 6.7.415.

Cluster Configuration

The cluster comprised an IBM x3650 M2 head node and 40 IBM iDataPlex DX360 M2 compute nodes connected on a fully-connected non-blocking QDR InfiniBand private network fabric. The file system was IBM GPFS on four IBM System x3650 M2 storage nodes connected to the QDR InfiniBand fabric. Each compute node contained two Intel Xeon x5550 processors running at 2.66 GHz with 24 GB of memory.

The Intel® Xeon® processor 5500 series architecture design draws on the benefits of hafnium-based Intel 45nm high-k metal gate silicon technology to deliver extremely high processing performance. Intel’s QuickPath Interconnect (QPI) delivers substantial increase in bandwidth from earlier architectures by incorporating an integrated DDR3 memory controller onto the processor die. Multi-level shared cache reduces latency to frequently used data, thereby improving performance and efficiency significantly. The processor can run two threads per core simultaneously, and exploits Intel’s Turbo Boost Technology to increase performance of both multi-threaded and single-threaded workloads by dynamically increasing the processor frequency based on the workload and number of active cores.

Performance Tests

Four test cases were provided by Milliman to illustrate the performance of MG-ALFA on an IBM System x cluster running Windows HPC Compute Server 2008. The size and computation times for the test cases varied significantly, with single-engine executions of the distributed computations ranging from under one hour for Test Case C2 to nearly 3 days for Test Case C1 using one core of a 2.66 GHz Intel Xeon x5550 processor. Relevant characteristics of the test cases are summarized in Table 1. For each test case, the table indicates the distribution type (i.e., the basis upon which each test case was divided into multiple independent tasks). MG-ALFA permits users to adjust the granularity of the parallel computation by specifying the task size (either scenarios per task or cells per task, depending on the distribution type). The table indicates the ranges of task sizes used in the performance tests.

Test Case	No. of Cells	No. of Scenarios	Distribution Type	Task Granularities Used
A	66,800	1	Liability Cell	200 or 500 cells per task
B	809	1,000	Scenario	5 or 10 scenarios per task
C1	13,140	100	Scenario	1, 2, 5, or 10 scenarios per task
C2	13,140	1	Liability Cell	100–500 cells per task

Table 1: Test Case Parameters

Performance Measurements

The primary measure of performance used in the performance tests is the elapsed time to complete a computation, measured in seconds. As noted in Table 1 above, several granularities were used for each test case, and the times reported here for a given number of MG-ALFA engines represent the smallest elapsed time over all the granularities tried. To characterize parallel scalability, a “speedup factor” is calculated as follows:

$$\text{Speedup Factor} = [\text{Elapsed Time using 1 engine}] \div [\text{Elapsed Time using } n \text{ engines}]$$

As described above, the individual tasks in a parallel MG-ALFA run are groups of simulation scenarios or liability cells. The task granularity and the target number of MG-ALFA engines are under user control by selecting values from menus in the desktop MG-ALFA interface. For many models, it is possible to set the task granularity so that the processing times (durations) for all tasks are approximately equal, in which case, it is reasonable to expect nearly linear scalability except for effects such as the following:

- *Task Distribution Overhead:* Depending on the way that MG-ALFA and the cluster are configured, there may be a significant amount of data copying or other overhead operations in setting up the tasks or returning the task results. This is particularly true when the task durations are short (under one minute). Sometimes, it may be appropriate to pre-stage data to a readily and rapidly accessible location, and MG-ALFA provides an option to avoid unnecessary data copying in this case. Frequently, the effect of the overhead may be mitigated by increasing the duration of the tasks (for example, by increasing the number of scenarios or cells per task), although the savings may be partially offset by other effects caused by the reduction in the total number of tasks.
- *Unbalanced Task Sizes:* If there is significant variation in the processing times for the individual tasks, then it may be very difficult keep all the MG-ALFA engines uniformly busy during the course of a parallel run. The impact of this imbalance may be somewhat difficult to predict since it depends on the number of engines, on the durations of each scenario or liability cell computation, and on the precise assignment of tasks to engines by the HPCS job scheduler. Typically, there is very little variation in time to run different scenarios, so task imbalance tends not to be a problem when using scenario distribution. However, in liability cell distribution, there may be significant variability in the time to process different model points (liability cells) due to dependencies on the age of the policyholders, the insurance periods, and variation in product features across the model points. Moreover, the insurance policies in the in-force file are often sorted by plan, issue date, and issue age so that consecutive policies in the file are likely to require similar computational effort, potentially leading to task imbalance if policies were assigned to tasks in the sorted order. To mitigate this possibility when using liability-cell distribution, MG-ALFA randomly assigns model points to the tasks.
- *“Tail” Effects:* Even when the task durations are approximately equal, if the total number of tasks is not divisible evenly by the number of engines, then some engines will be idle while all the rest are working on their final tasks. (In the worst case, all but one engine will be idle.) The latest MG-ALFA interface permits the user to specify the number of scenarios/cells per task and the target number of engines, so it should be possible to mitigate tail effects in most cases. Moreover, when there is insufficient work available for all the engines, the current version of MG-ALFA attempts to

give cores back to the Job Manager so that they may be used for other jobs. (This does not improve the performance of the job in question, but may improve the overall throughput and efficiency of the cluster as a whole.)

- *Local Resource Contention:* While the scenarios and tasks are completely independent from one another in a mathematical sense, the computations running on each node do compete with one another for memory, I/O, network capacity, and other resources. This may cause them to run less efficiently as a group than they would have run individually as standalone tasks. This effect may be mitigated by providing sufficient memory (at least 2 GB per core) and using high-performance disk configurations (either a parallel file system like IBM's GPFS or a multi-disk RAID configuration on each node).
- *Global Resource Contention:* With all the tasks requiring approximately equal amounts of processing time, there is some danger that the ending times of the tasks will be synchronized in a way that causes delay between the completion of a task on a core and the start of the next task on the same core. If the idle time due to this effect is significant in the aggregate, it may impact the observed scalability. This effect is more likely to be significant if the total number of engines is very large or the tasks are very short.
- *Serial Portions of MG-ALFA:* The times reported below reflect only the time to complete the embarrassingly-parallel portion of the computation. In some models there may be additional pre- and post-processing performed serially by MG-ALFA on the desktop, which could impact the apparent scalability were it included in the times. (For example, neither initial task creation nor result communication/collation are included in the elapsed times listed.) This effect is most significant with a large number of short tasks.

With the current versions of MG-ALFA and HPCS, avoidance of the effects impeding scalability is largely the responsibility of the user. For example, to avoid tail effects, users should strive to make the number of tasks divisible evenly by the target number of engines requested, particularly when the quotient is small. As an illustration, for a job containing 200 equal-duration tasks, the elapsed time using 190 engines will be much closer to the elapsed time using 100 engines than to the elapsed time using 200 engines.

Test Results and Tips for Optimizing Performance

The combination of MG-ALFA, Windows HPC Server 2008, and IBM BladeCenter or System x clusters delivers excellent performance, demonstrating near-linear scalability in all cases. The elapsed times and speed-up factors for the four test cases are presented in Table 2 on page 11 and plotted in Figures 2–5 on pages 12–13. These test cases were ideal fits to the “grid-style” parallelism of MG-ALFA, and performance depended primarily on the computational capabilities of the cluster nodes, rather than on I/O capabilities or the speed of the network interconnect. Only a relatively small amount of data had to be copied between the head node (where the MG-ALFA client ran) and the compute nodes, so the impact of the network speed was not especially large. In addition, the model was small enough so that there were no memory constraints on the compute nodes, where 3 GB of memory per engine was available.

There is insufficient space here to discuss many details of the performance testing process. However the results did demonstrate the importance of selecting task granularities so that the overhead related to task start-up and shut-down was minimal in comparison to the duration of task computation. In addition, for test cases like Test Cases B and C1, for which the tasks are generally nearly equal in duration, the results made clear the value of avoiding tail effects by correlating the task granularities with the number of engines, particularly when only a small number of tasks per engine would be available. Without such correlation, many of the engines could be idle during the final portion of the computation, wasting a significant portion of the total computational power committed to the job by the Job Manager. (It is worth noting that MG-ALFA attempts to mitigate this phenomenon by shutting down unneeded engines.)

The Bottom Line

Competitive pressures and regulatory changes in the life insurance industry increasingly require the ability to analyze large actuarial models that include a thousand or more scenarios, and to perform that analysis more frequently. Gone are the days when users could expect to run such models effectively on their desktop workstations, but until now, better solutions required significant infrastructure investments and were often poorly integrated with preferred end-user desktop computing environments. This made it extremely difficult for insurance industry firms to move to high performance solutions that could meet their modeling demands.

Offerings from IBM, Microsoft, and Milliman now make it practical and affordable for firms in the insurance industry to adopt HPC solutions for actuarial modeling. The combination of Microsoft Windows HPC Server 2008 running on IBM clusters provides an ideal infrastructure for running modeling software such as Milliman's MG-ALFA, which has been enhanced with an easy-to-use parallel computing capability. As demonstrated in this white paper, this combined solution can deliver extremely high performance that can be scaled up to meet a variety of end-user and IT demands. Moreover, since the end-user experience is almost identical to what it is with the desktop version of MG-ALFA, the solution should fit naturally with current operating procedures and industry best practices.

Of course, migration to in-house HPC solutions may not be simple. There may be significant initial infrastructure investments, as well as ongoing costs for space, power, and cooling. In addition, actuarial workloads tend to vary on a cyclical basis, so providing sufficient capacity to handle peak loads may mean that significant resources lie idle the rest of the time. IBM's Computing on Demand offering avoids these issues by providing flexible access to a secure, fully managed HPC cluster infrastructure that can be applied almost immediately to dramatically improve actuarial modeling computations. Users pay only for the time and resources required to complete their jobs. The best way to get started with MG-ALFA in an HPC environment and see how it can benefit your actuarial modeling is to take an MG-ALFA test drive in an IBM Computing on Demand cloud center. Basic CoD test drives are available from IBM, or, if MG-ALFA application support is desired, you may request an MG-ALFA CoD test drive from Milliman. For more information on MG-ALFA CoD test drives, see Appendix A.

Results for IBM® System x™ iDataPlex™ dx360 M2 Cluster with Intel® Xeon® x5550 Processors								
Engines Used	Test Case A		Test Case B		Test Case C1		Test Case C2	
	Time (Seconds)	Speedup Factor	Time (Seconds)	Speedup Factor	Time (Seconds)	Speedup Factor	Time (Seconds)	Speedup Factor
1	10,380	1.00	29,185	1.00	239,975	1.00	2,511	1.00
2	5,096	2.04	14,678	1.99	120,871	1.99	1,259	1.99
4	2,638	3.93	7,736	3.77	64,777	3.70	736	3.41
5	2,095	4.95	6,167	4.73	51,427	4.67	591	4.25
10	1,193	8.70	3,098	9.42	26,248	9.14	302	8.31
20	605	17.16	1,555	18.77	13,190	18.19	167	15.04
30	419	24.77	1,124	25.97	10,475	22.91	128	19.62
40	337	30.80	808	36.12	7,767	30.90	102	24.62
50	267	38.88	647	45.11	5,303	45.25	88	28.53
100	174	59.66	328	88.98	2,684	89.41	57	44.05
132*							39*	64.38*
170	132	78.64						
200			169	172.69				

*MG-ALFA automatically reduced the number of engines from the specified number of 140 to 132 because Test Case C2 contained only 132 tasks when using a granularity of 100 cells per task.

Table 2: Performance Results for MG-ALFA Test Cases on an IBM iDataPlex Cluster

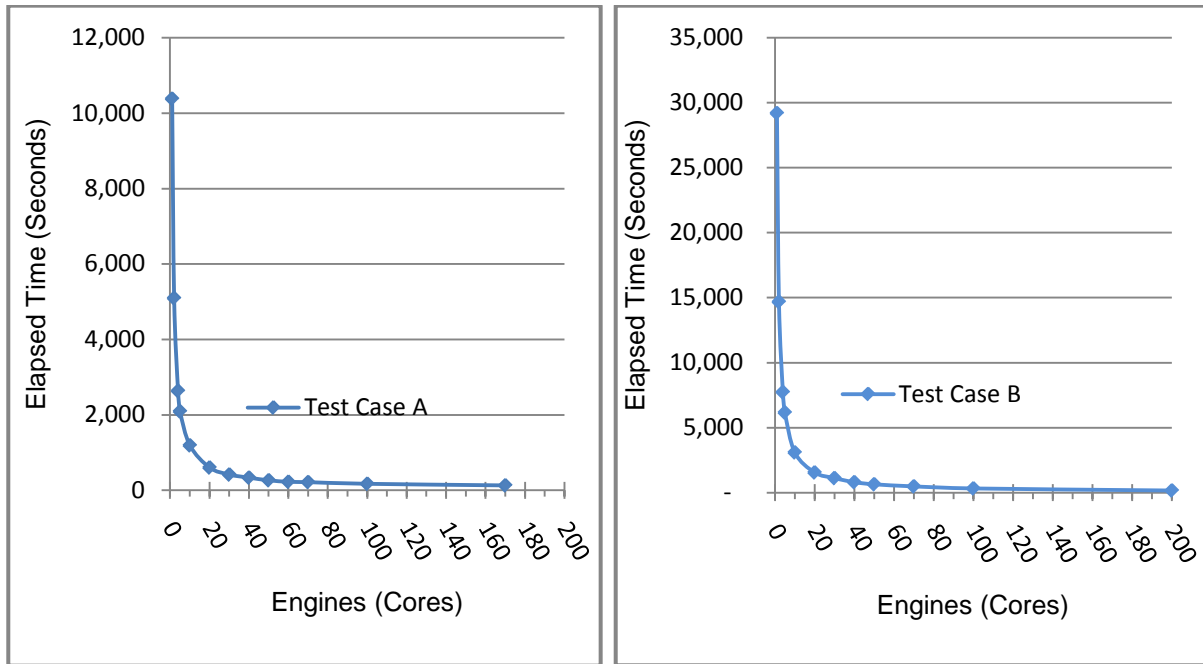


Figure 2: Elapsed Times for Test Case A and Test Case B

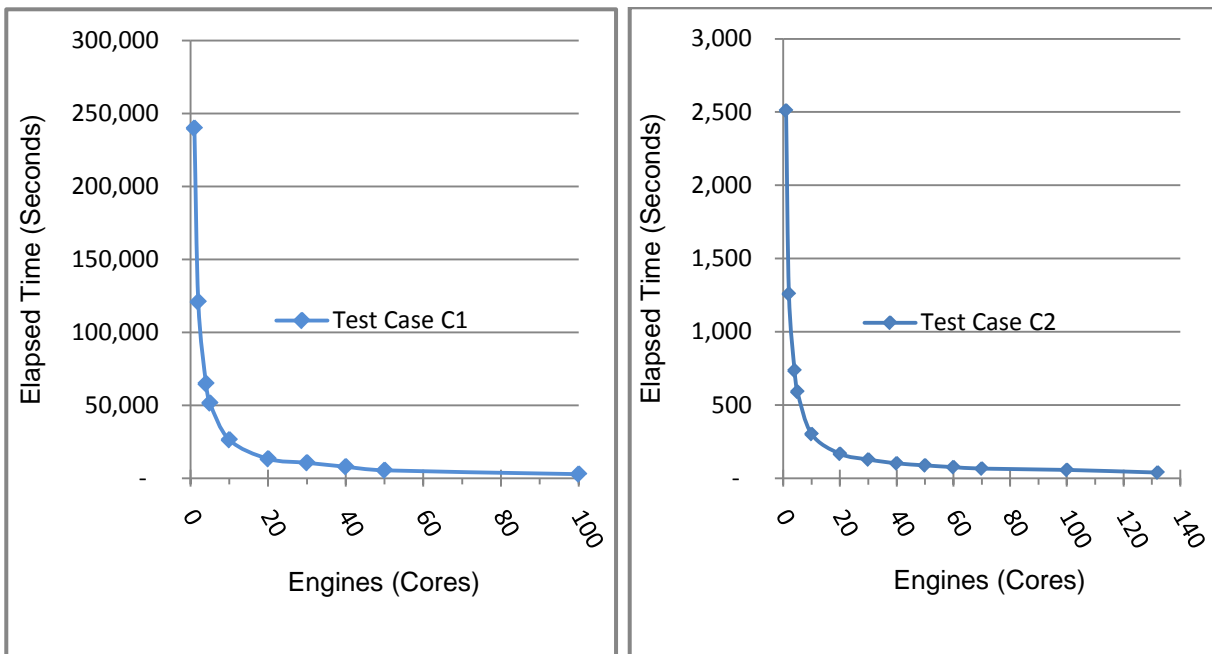


Figure 3: Elapsed Times for Test Case C1 and Test Case C2

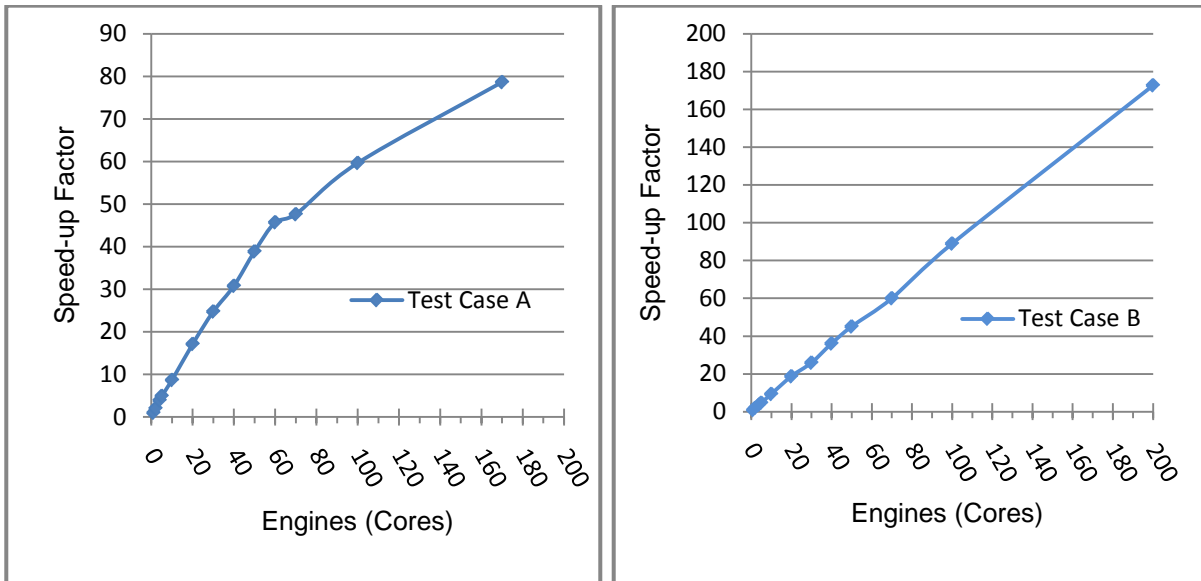


Figure 4: Speed-up Factors for Test Case A and Test Case B

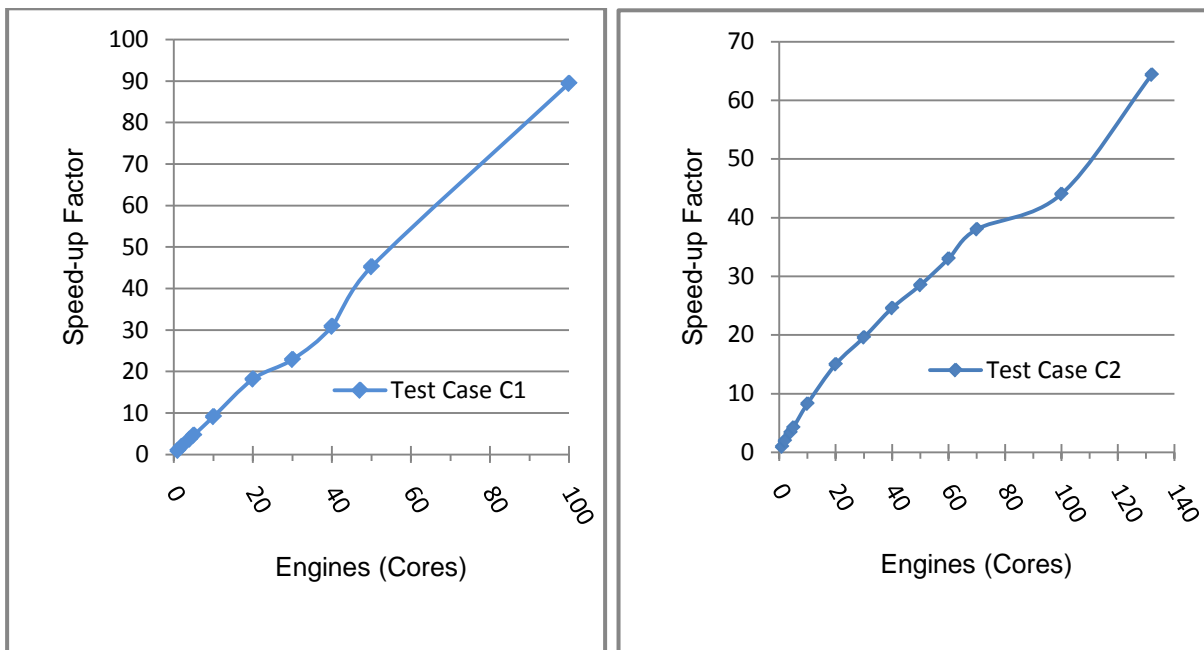


Figure 5: Speed-up Factors for Test Case C1 and Test Case C2

APPENDIX A: MG-ALFA Test Drive via IBM Computing on Demand

IBM or Milliman can offer a test drive of MG-ALFA in an IBM CoD cloud center. To sign up for a basic IBM CoD test drive, please register at ibm.com/cloud/testdrive. If MG-ALFA application support is desired, please request a MG-ALFA CoD test drive from Milliman via the [Milliman MG-ALFA cloud website](#).

Once you've registered and executed a test-drive agreement, IBM will schedule your test drive on a cluster running Microsoft Windows HPC Server 2008 that has been preconfigured with the MG-ALFA software and some sample datasets. (Of course, you'll be able to run your own datasets as well.)

For test drives, IBM's CoD centers provide a secure connection to the test drive clusters using an SSL or software VPN client connection. The IBM CoD Support Center can provide details.

After establishing a secure connection to the CoD Center, you connect to your test drive cluster using Windows Remote Desktop Connection. IBM's test drive welcome package will include both the network address and account information required to do this.

When you login to the cluster, you'll be on the head node of the Windows HPC Server 2008 cluster, which is where you'll run the MG-ALFA desktop client. You'll see a shortcut on the desktop to MG-ALFA, version 6.7. Before running MG-ALFA, however, it's a good idea to start the Windows HPC Cluster Manager application by selecting it off of the Start menu. The Cluster Manager enables you to monitor the status and performance of the cluster nodes and to manage the jobs you'll run through MG-ALFA.

If you wish to upload any data for your MG-ALFA test drive, you may do it by using the remote copy and paste functionality of Windows Remote Desktop Connection. On your local desktop, simply right click on the file you wish to upload and select "Copy" from the pop-up menu that appears. Then, on the cluster head node, paste the file wherever you wish. In order to use the files for parallel runs, you must share the folder containing them so that the files are accessible from all the cluster nodes.

Before running MG-ALFA for the first time, you should make certain that the following folders have been created on the head node:

<code>C:\MGALFA\TOOLS\CCS</code>	Folder containing an MG-ALFA configuration file
<code>C:\MGALFAUSER</code>	Work folder used to share files with the compute nodes

The `MGALFAUSER` folder must be shared so that it is accessible from all the cluster nodes. In the `CCS` folder, there should be a file named `ms.ini` that should contain the following two lines:

```
[main]

Server=<name or IP address of the head node>
```

To run the MG-ALFA desktop client on the cluster head node, double click on the MG-ALFA shortcut on the desktop. The main MG-ALFA window will open. The first time you run MG-ALFA, you should check to make sure that the MG-ALFA configurations are correct. Select `Config` from the `Options` menu, which will open a settings window. For each setting, there is a button that, when clicked, pops up the menu or dialog box used to change the setting. The following settings are important:

1. The `SDPAvailable` option should be set to `MSCluster`.
2. The value of the `SDPfolder` setting should be `C:\MGALFA\TOOLS\CCS`.
3. The value of the `WorkFolder` setting should be the UNC file name of the work folder (such as `\\<head_node_name>\MGALFAUSER\`).

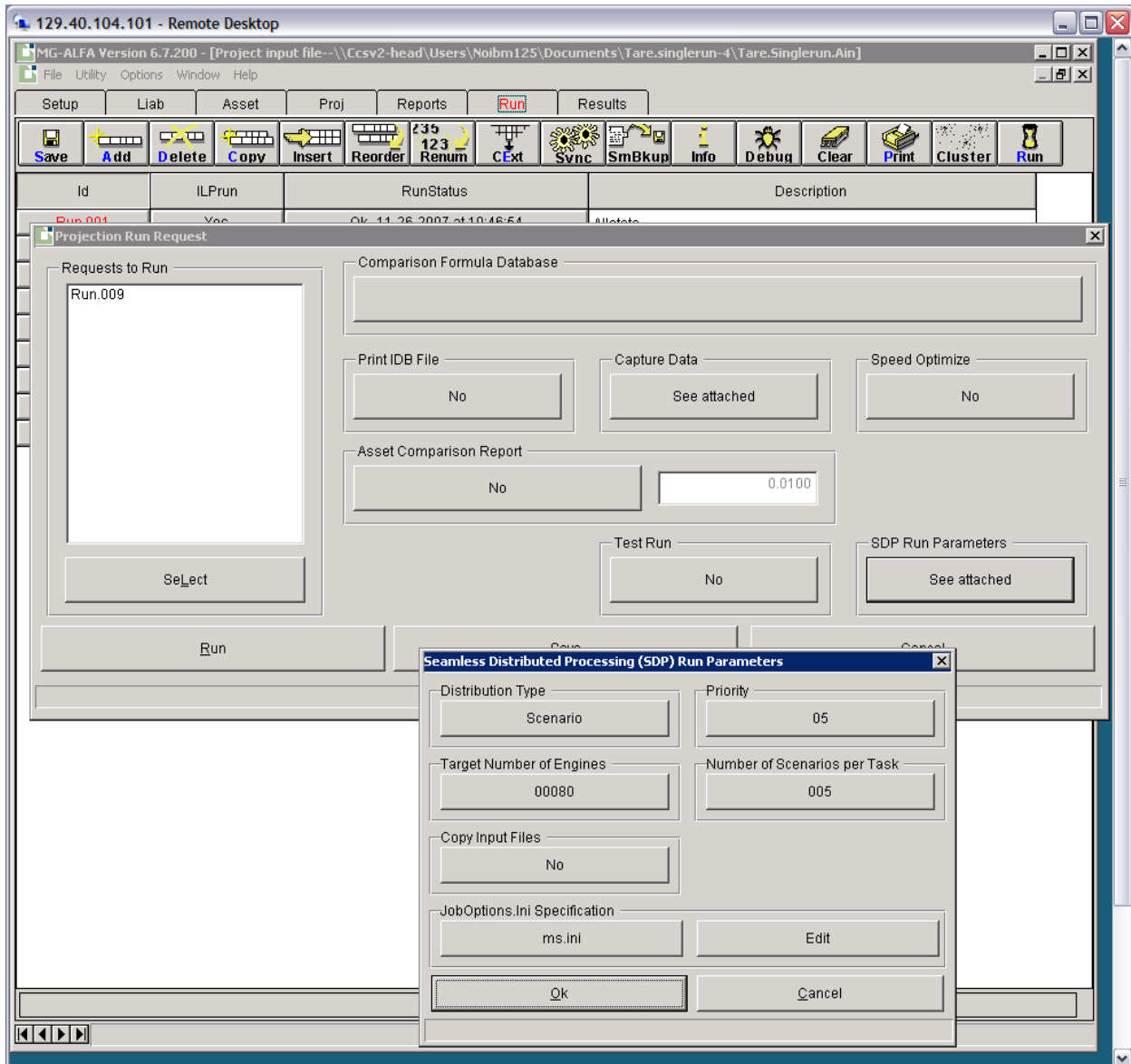


Figure 6: MG-ALFA Screenshot displaying SDP Run Parameters settings window

Once the configuration settings are set correctly, click on the OK button to close the settings window. Now you can select Open from the File menu to open an input file (a file with an “AIN” extension). In order to make use of the cluster, you need to open the input file using its UNC file name (such as “\\<head_node_name>\c\users\userid\documents\testcase\file.ain”) rather than its local file name. After it opens the file, MG-ALFA will display the contents of the Setup tab.

To make a parallel run, click on the Run tab and then on the Run button in the toolbar. This will open a “Projection Run Request” window (see Figure 6). Now click on the “SDP Run Parameters” button (labeled “See attached”), which pops up the “Seamless Distributed Processing (SDP) Run Parameters” window. The following parameters should be set here by clicking on the appropriate button to pop up a menu from which the values may be selected:

1. *Distribution Type*: This setting controls the method used to subdivide the full model into multiple tasks. The choices are:
 - a. `Na`: Serial run on the head node (no parallelism);
 - b. `Scenario`: Each task includes a number of scenarios;
 - c. `LiabCell`: Each task includes a number of liability cells (portfolio instruments or policies)
2. *Number of Scenarios per Task (or Number of Liability Cells per Task)*: This parameter sets the task granularity. The value is selected from a pop-up menu of choices. Reasonable defaults are 10 scenarios per task or 500 liability cells per task, but experimentation to adjust the granularity may lead to more efficient choices.
3. *Target Number of Engines*: This parameter is the target number of engines to request from the HPCS Job Manager. Each engine will use one core on one compute node, and the Job Manager will pack the requested cores into as few nodes as possible. If the number of engines requested exceeds the number of tasks in the entire run, then MG-ALFA will automatically reduce the number requested to equal the number of tasks.
4. *Copy Input Files*: This setting controls the way that input files are communicated to the compute nodes. In general, the default choice should be `No`, in which case the compute nodes read the file from the head node.
5. *JobOptions.ini Specification*: This parameter should be set to `ms.ini` by selecting that value from the pop-up menu. This links to the `ms.ini` file mentioned above.
6. *Edit*: This button opens a text editor that may be used to change the contents of the `ms.ini` file should that be necessary.

When all parameters have been set, click `OK` to save the settings. Then click the `Run` button to start the run. As described above on page 6, MG-ALFA will then go through a number of processing steps, eventually leading to a job submission to the HPCS Job Manager. Once the job has been submitted, MG-ALFA will display a status window indicating the job status. When the job is completed, MG-ALFA will assimilate the results just as if the computation had occurred in a local serial run.

To assess the parallel performance, it is recommended that you use the times available in the Job Management section of the HPCS HPC Cluster Manager application. (The “Setup” page in MG-ALFA’s “RunStatus” information will also provide timings, but those times include time spent waiting in the Job Manager queue, so they may not provide a good indicator of actual computation time.) A listing of your jobs may be displayed in the HPC Cluster Manager by clicking on the “Job Management” button and then on the “My Jobs” navigation entry. By clicking on the line for a particular job, you will see corresponding information in a tabbed window pane at the bottom of the screen. The “Activity Log” tab will give overall timing information for the job, including start and stop times and node/core usage. The “Tasks” tab will provide detailed information about the execution of each task.

To illustrate the type of performance that you might expect to see in your MG-ALFA test drive, two of the white paper test cases were run on an HPCS cluster similar to one that you could use at a CoD center. The cluster comprised IBM BladeCenter HS21 nodes connected via a Gigabit Ethernet private network. Each node contained two dual-core Intel Xeon 5160 Woodcrest processors running at 3.0 GHz, 8 GB of memory, and a single SAS disk. The tests were run using MG-ALFA Version 6.7.200.

Table A-1 contains the results, which, like those on the iDataPlex cluster, demonstrate excellent scalability (as illustrated graphically in Figure A-1). The primary distinction is that MG-ALFA runs significantly faster on Intel’s Nehalem processors than it does on the older Woodcrest processors, most likely because of the improved memory architecture on the Nehalem processors.

For additional information about MG-ALFA operation, users should consult the MG-ALFA documentation installed on the CoD cluster. For additional information about Microsoft Windows HPC Server 2008, users should use the help information in the HPC Cluster Manager application or visit the Microsoft website at <http://www.microsoft.com/hpc/en/us/default.aspx>.

Results on an IBM CoD Cluster				
Engines Used	Test Case B		Test Case C1	
	Time (Seconds)	Speedup Factor	Time (Seconds)	Speedup Factor
1	33,045	1.00	307,905	1.00
2	17,409	1.90	165,658	1.86
4	8,698	3.80	82,745	3.72
5	6,951	4.75	66,414	4.64
10	3,487	9.48	33,313	9.24
20	1,748	18.90	16,723	18.41
30	1,267	26.08	13,265	23.21
40	910	36.31	9,976	30.86
50	702	47.07	6,718	45.83
100	390	84.73	3,388	90.88
200	191	173.01		

Table A-1: Performance Results for MG-ALFA Test Cases on an IBM CoD Woodcrest Cluster

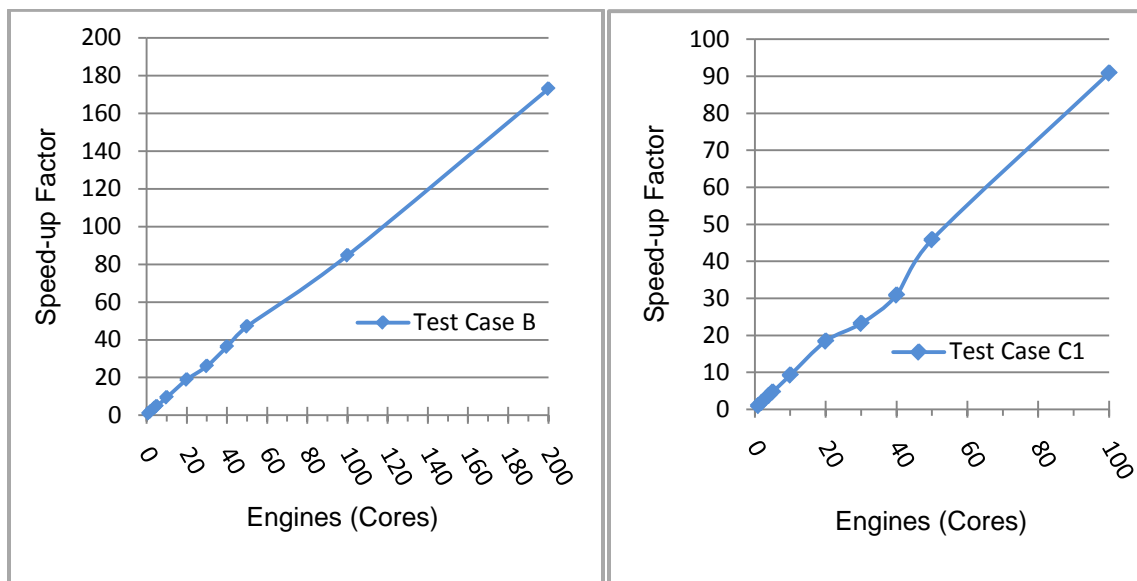


Figure A-1: Speed-up Factors for MG-ALFA Test Cases on an IBM CoD Woodcrest Cluster

APPENDIX B: Additional Information

Additional information about the companies, products, and technologies discussed in this white paper is available on-line at a number of websites. Suggested sites are listed here.

- For information about IBM HPC solutions, visit the following sites:
 - HPC solutions for financial services, insurance, and banking: http://www-03.ibm.com/systems/deepcomputing/solutions/ind_financial.html
 - IBM System x and BladeCenter servers with Microsoft technologies: <http://www-03.ibm.com/systems/x/solutions/os/windows/index.html>
 - IBM Cluster 1350: <http://www-03.ibm.com/systems/clusters/hardware/1350/>
 - IBM Computing on Demand: <http://www-03.ibm.com/systems/services/dccod/>
 - IBM CoD Test Drives: <http://www-03.ibm.com/systems/deepcomputing/cod/testdrive.html>
- For information about Microsoft Windows HPC Server 2008, visit the following sites:
 - Home page: <http://www.microsoft.com/hpc/en/us/default.aspx>
 - Features: <http://www.microsoft.com/hpc/en/us/features.aspx>
 - Product details: <http://www.microsoft.com/hpc/en/us/product-details.aspx>
 - Product documentation: <http://www.microsoft.com/hpc/en/us/product-documentation.aspx>
 - Technical white papers: <http://technet.microsoft.com/en-us/library/cc510343%28WS.10%29.aspx>
- For information about Milliman's MG-ALFA, visit the following sites:
 - Home page: <http://www.milliman.com/mg-alfa>
 - Cloud Computing: <http://www.milliman.com/expertise/life-financial/products-tools/mg-alfa/cloud-computing.php>



© Copyright IBM Corporation 2009

IBM Systems and Technology Group
Route 100
Somers, NY 10589

Produced in the United States
October 2009
All Rights Reserved

IBM, the IBM logo, BladeCenter, System x, iDataPlex, Power, AIX 5L are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries or both. A full list of U.S. trademarks owned by IBM may be found at: <http://www.ibm.com/legal/copytrade.shtml>

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries or both.

Milliman and MG-ALFA are trademarks of Milliman Inc. in the United States, other countries or both.

Intel, the Intel Logo, Intel Inside (logos) and Xeon are trademarks of Intel Corporation in the United States, other countries or both.

Linux is a trademark of Linus Torvalds in the United States, other countries or both.

Other company, product and service names may be trademarks of others.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

IBM hardware products are manufactured from new parts, or new and used parts. In some cases, the hardware product may not be new and may have been previously installed. Regardless, our warranty terms apply.

All performance information was determined in a controlled environment. The results obtained in other operating environments may vary.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. All information in these materials is subject to change without notice.

ALL INFORMATION IS PROVIDED ON AN "AS IS" BASIS, WITHOUT ANY WARRANTY OF ANY KIND.

The IBM home page on the Internet can be found at ibm.com