Considerations for LS-DYNA Workflow Efficiencies in an HPC Linux Environment

Stanley Posey
HPC Applications Development
SGI, Mountain View, CA
sposey@sgi.com

Abstract

Manufacturing industry and research organizations continue to increase their investments in structural analysis and impact simulations such that the growing number of LS-DYNA users continues to demand more from computer system environments. These LS-DYNA workflow demands typically include rapid single job turnaround and multi-job throughput capability for users with diverse application requirements in a high-performance computing (HPC) environment.

Additional complexity arises from the need for many LS-DYNA HPC environments to coexist with other computer aided engineering (CAE) software for a variety of multi-physics and multi-scale structural and computational fluid dynamics analyses that all compete for the same HPC workflow resources. For today's economics of HPC, these resources such as CPU cycles, memory, system bandwidth and scalability, storage and I/O, and file and data management – must deliver the highest levels of CAE productivity and HPC reliability that is possible from a Linux platform environment.

This presentation examines workflow efficiencies of CAE simulations for relevant applications in LS-DYNA for an HPC Linux platform developed by SGI. LS-DYNA modeling parameters such as model size, element types, schemes of implicit and explicit (and coupled), and a variety of simulation conditions can produce a wide range of computational behavior and data management requirements, such that careful consideration should be given to how system resources are configured, deployed, and allocated to meet increasing user demands.

The HPC system technology of the SGI® Altix™ clusters and servers, based on Linux® and Itanium® 2 from Intel®, have demonstrated both LS-DYNA turnaround and throughput achievement that includes industrial-sized examples. In addition, SGI simulation data management technology of HPC file systems and data storage tools, are providing the LS-DYNA workflow management necessary to maximize user productivity, and enable a user roadmap of increasing LS-DYNA modeling fidelity.

Introduction

Mechanical design and manufacturing organizations increasingly rely on high-performance computing (HPC) technology and mechanical computer-aided engineering (MCAE) applications to drive innovation in product development. Industries such as automotive, aerospace, turbomachinery, and others are challenged with an increasing need to reduce development cycles, while satisfying global regulations on safety, environmental impact and fuel efficiency. They must also appeal to demands for high quality, well-designed products in a competitive business environment.

Continuing technology advances in finite element analysis (FEA) provide industry with a design aid that is a relevant step towards achieving their product development goals. Historically, FEA simulation provided limited value as an influence on industrial design owing to excessive modeling and solution times that could not meet conventional development schedules. During the 1980's vector architectures offered greatly improved FEA simulation turn-around, but at a very high cost. Scalable RISC computing introduced in the 1990's narrowed this gap of cost-performance.

Recent advancements in parallel computing have demonstrated that vector-level performance can be easily exceeded with proper implementation of parallel FEA algorithms for distributed shared-memory servers like the SGI Altix 4700 and clusters like the SGI Altix 1330. Perhaps even more appealing is that the increased performance is offered at a fraction of the cost. These trends have influenced recent increased investments by users of FEA technology throughout a range of industries.

The HPC paradigm, however, has been undergoing a change in recent years thanks to a proliferation of Linux-based scalable systems. Such systems equipped with Intel processors and some, such as the SGI Altix 4700 that offer an inexpensive yet innovative memory design, can deliver a level of performance that exceeds vector and most RISC systems in nearly all aspects. For example, all conventional clusters are capable of processing thousands of MIPS and hundreds of MFLOPS, a level well beyond the single digit performance in the mid-1980's RISC when it was first introduced.

With release of LS-DYNA 940 during 1999, the first substantial parallel capability for explicit FEA simulation provided the industry with a migration path from expensive vector systems to more economical scalable systems. A distributed memory parallel (DMP) programming model is used for LS-DYNA in order to leverage the advantage of these contemporary scalable architectures. This paper examines efficient implementation of LS-DYNA for both uni-processor and parallel considerations for scalable 64-bit Linux servers and clusters.

HPC Characteristics of LS-DYNA

Finite element analysis software LS-DYNA from Livermore Software Technology Corporation (www.lstc.com) is a multi-purpose structural and fluid analysis software for high-transient, short duration structural dynamics, and other multi-physics applications. Considered one the most advanced nonlinear finite element programs available today, LS-DYNA has proved an invaluable simulation tool for industry and research organizations who develop products for automotive, aerospace, power-generation, consumer products, and defense applications, among others.

LS-DYNA simulations for the automotive industry include vehicle crash and rollover, airbag deployment and occupant response. For the aerospace industry, LS-DYNA has ability to simulate bird impact on airframes and engines and turbine rotor burst containment, among others. Additional complexities arise from simulations of these classes since they often require predictions of surface contact and penetration, modeling of loading and material behavior, and accurate failure assessment.

From a hardware and software algorithm perspective, there are roughly three types of LS-DYNA simulation "behavior" to consider: implicit and explicit FEA for structural mechanics, and computational fluid dynamics (CFD) for fluid mechanics. Each discipline and associated algorithms have their inherent complexities with regards to efficiency and parallel performance, and also depending upon modeling parameters.

The range of behavior for the three disciplines that are addressed with LS-DYNA simulations, highlights the importance of a balanced HPC system architecture. For example, implicit FEA for static load conditions requires a fast processor for effective turnaround, in contrast to dynamic response, which requires high rates of memory and I/O bandwidth with processor speed as a secondary concern. In addition, FEA modeling parameters such as the size, the type of elements, and the load condition of interest all affect the execution behavior of implicit and explicit FEA applications.

Explicit FEA benefits from a combination of fast processors for the required element force calculations and a high rate of memory bandwidth necessary for efficient contact resolution that is required for nearly every structural impact simulation. CFD also requires a balance of memory bandwidth and fast processors, but benefits most from parallel scalability. Each discipline has inherent complexities with regard to efficient parallel scaling, depending upon the particular parallel scheme of choice.

Implementations of both shared memory parallel (SMP) and distributed memory parallel (DMP) have been developed for LS-DYNA. The SMP version exhibits moderate parallel efficiency and can be used with SMP computer systems only, while the DMP version, exhibits very good parallel efficiency. This DMP approach is based on domain decomposition with MPI for message passing, and is available for heterogeneous compute environments such as shared memory parallel systems or clusters.

Most parallel MCAE software employ a similar DMP implementation based on domain decomposition with MPI. This method divides the solution domain into multiple partitions of roughly equal size in terms of required computational work. Each partition is solved on an independent processor, with information transferred between partitions through explicit message passing software (usually MPI) in order to maintain the coherency of the global solution.

LS-DYNA is carefully designed to avoid major sources of parallel inefficiencies, whereby communication overhead is minimized and proper load balance is achieved. Another advantage for LS-DYNA is the use of an SGI-developed MPI that is NUMA-aware and transparent to the user. This MPI further reduces communication overhead when scaling to a large number of processors, which is achieved by a reduction in latency that is improved over public-domain MPI such as MPICH and LAM.

SGI Altix HPC Architecture

SGI servers and clusters are based on a cache-coherent non-uniform memory access (NUMA) multiprocessor architecture that is a breakthrough implementation of conventional shared memory architectures. The SGI NUMA architecture distributes memory to individual processors through a non-blocking interconnect design, in order to reduce latencies that inhibit high bandwidth and scalability. At the same time, all memory is globally addressable, meaning memory is physically distributed but appears logically as a shared resource to the user, to accommodate high-fidelity MCAE modeling and simplify MCAE algorithm development.

The motivation for this distributed shared memory NUMA approach evolved at SGI when conventional shared-bus architectures began to exhibit high-latency bottlenecks as the number of processors increased within a single system shared-memory image. During this same time, non-coherent distributed memory architectures started to emerge, but the programming of applications for message passing in such an environment was considered too complex for commercial success.

This SGI NUMA architecture was introduced in the SGI Origin 2000 server in 1995 and later advanced with the SGI® NUMAflex™ modular design concept of the SGI Origin 3000 series. A single image SGI Origin 3900 system could offer up to 512 processors and can expand to 1 TB of memory, and at the time was the largest shared memory scalable system available in industry. It is based on an SGI proprietary platform of the IRIX® operating system, the MIPS® microprocessor.

Since January 2003, this same NUMAflex architecture of the SGI Origin series has been available in the SGI Altix, an HPC platform based on 64-bit Linux and the Itanium 2 microprocessor from Intel. At its launch, the SGI Altix 3000 was the first Linux-based HPC system that retained supercomputing features such as global shared-memory for a single image, yet also offered cluster capability. The SGI Altix is an open standards-based platform that combines industry leading HPC system technologies.

The SGI Altix is built-up from a number of component modules, or bricks, most of which are similar with the SGI Origin family. The C-brick (compute brick) is the module that customizes the system to a given processor architecture. The SGI Altix C-brick consists of four Intel Itanium 2 and corresponding memory up to 8GB, and a connection to a portion of an I/O subsystem. The hub interface to the C-brick is the distributed memory controller and C-bricks are connected together via the R-brick, which is the NUMAflex router module.

The maximum configuration of a single node of the latest implementation, the Altix 4700, is 512 processors in a blade form factor, and up to 128 TB of shared memory within each single Linux OS-image node. These individual 512 processor nodes can be clustered with a choice of scalable interconnect networks, including the proprietary SGI® NUMAlink™ interconnect technology, to much larger system configurations of up to 2,048 processors and a total of 512 TB of memory – a configuration that offers the highest levels of compute resources available in the industry.

The SGI® Altix® 1330 cluster provides SGI's Altix® architecture in a fully-integrated cluster system. SGI Altix 1330 provides many features available in the SGI Altix 4700 —yet in a cost-effective cluster platform. The Altix 1330 system design leverages the best of large node capability plus cluster capacity—with nodes that can scale to 16 processors and 128GB of memory. Several interconnect and system management tools from SGI, Voltaire and Scali, to name a few, that provide performance and efficiency in support of very large configuration clusters.

The SGI Altix is binary compatible with industry-standard 64-bit Linux kernel, and offers a choice of popular distributions such as Novell® SUSE LINUX and Red Hat®. In addition, SGI offers differentiated middleware and other functionality to enhance technical HPC workloads in a software module called ProPack -- a set of user tools that ride on top of Linux. ProPack is

similar to other commercial software packages and is used to boost the performance of Linux and user applications on the SGI Altix, and not to alter Linux itself.
Various other system level tools such as the Intel compilers, the SGI-developed SCSL library, commands runon, dplace, and others are available on the SGI Altix to help LS-DYNA achieve its performance goals. The general principle of HPC performance tuning for the SGI Altix is to ensure compiler software pipelining is invoked so that efficient instruction scheduling occurs, which provides good processor, cache, and memory locality of data during execution.

The typical practice of an LS-DYNA user environment is to combine moderate scalability of up to 32 processors for a single job with multi-job throughput, in a mix that provides the greatest overall cost benefits of a particular application. The SGI Altix combines the HPC technologies that offer industry and manufacturing research organizations a high-availability, non-degrading, and efficient LS-DYNA application environment that ensures turnaround and throughput are delivered in support of hundreds of simultaneous users with a mix of MCAE disciplines.

Performance Considerations of LS-DYNA

Performance and parallel efficiency of any MCAE software has certain algorithm considerations that must be addressed. The fundamental issues behind parallel algorithm design are well understood and described in various research publications. For grid-based problems such as the numerical solution of partial differential equations, there are four main sources of overhead that can degrade ideal parallel performance: 1) non-optimal algorithm overhead, 2) system software overhead, 3) computational load imbalance, and 4) communication overhead.

Parallel efficiency for LS-DYNA is dependent upon among others, MPI latency, which is determined by both the specifics of system architecture and the implementation of MPI for that system. Since system architecture latency is determined by design of a particular interconnect, overall latency improvements can only be made to the MPI implementation. Modifications to the MPI software to ensure "awareness" of a specific architecture are the only way to reduce the total latency and subsequently the communication overhead. Improvements to architecture-awareness of MPI is a common development for several system vendors including SGI.

Parallel efficiency bottlenecks for LS-DYNA on SGI Altix systems are identified as MPI latency and non-enforcement of processor-memory affinity (data placement) as the most dominant. The latency problem was solved by invoking SGI NUMA-aware MPT library, which appears in LS-DYNA for SGI Altix systems. The data placement concern is related to a feature of the NUMA architecture and is addressed through implementation of the SGI dplace set of tools that are provided on the SGI Altix.

Specifically for structural FEA simulations, they often contain a mix of materials and finite elements that can exhibit substantial variations in computational expense, which may create load-balance complexities. The ability to efficiently scale to a large number of processors is highly sensitive to load balance quality. For example, the crash worthiness of automotive vehicles exhibit these characteristics.

An additional consideration for improved load balance is efficient use of the domain decomposer. The automotive model in Figure 1. illustrates the results of a geometric-based RCB decomposition with use of (left) default parameters and (right) use of an improved scheme that exhibits better overall performance.

Figure 1. LS-DYNA Decomposition Results for Default (left) and Modified (right) RCB

For this particular model and partitioning strategy, the modified RCB achieved an improvement of 37% over the default RCB partitioning for 8 processors, and this advantage grew to 55% for 16 processors. This is due to the fact that the modified RCB provides better data partitioning with a more even distribution of processor workloads and less communication between processors.

As a result of collaboration with joint technologies of HPC systems from SGI and FEA software from LSTC, LS-DYNA when executed on an SGI Altix series system, often exhibits linear scalability as high as 96 processors for various industrial-sized automotive vehicle and aerospace applications. This substantial savings in simulation turnaround time allows additional studies to be performed towards optimization of a vehicle's crashworthiness and safety, or leads to better understanding of gas turbine blade-out evolution and containment.

LS-DYNA parallel scalability on the SGI Altix has demonstrated the abilities of a moderately configured SGI Altix system to reduce the compute time of an 800 K element model of 150 milliseconds, to less than 1 hour. The system contained 96 Itanium 2 processors from Intel, in globally shared memory arrangement with 32 GB of memory. The model for this test was car-to-car of 3 automotive vehicles that appears on the web site www.topcrunch.org that is supported by DARPA.

The use of 150-milliseconds ensures that the model considers the effect of contact in its ability to scale. During the first 5 milliseconds of simulation there are essentially no contacts, and such a performance evaluation at this state of a simulation would not provide a true test of parallel scalability for a fully converged simulation.

Evaluation of an vehicle's crashworthiness is currently the fastest growing application for MCAE simulation in the automotive industry. Proper crash management and occupant safety is a mandate by local governments, but it’s also viewed as a competitive advantage by many automotive developers. Performance improvements continue with LS-DYNA such that parallel scalability keeps pace with growing demand.
Similarly, an aerospace application for design of gas turbine engines for aircraft, has utilized the parallel scalability of LS-DYNA on a 128-processor SGI Altix system to reduce from weeks to days, the time it takes to complete a 4MM-element model for blade-out simulation. Parallel efficiency for SGI Altix for this model exceeded linear scalability, demonstrating the benefit of the SGI NUMAflex architecture for Linux and Intel Itanium 2 microprocessors.

Additional developments between LSTC and SGI applications engineering include an enhanced I/O scheme that significantly improves overall model turnaround in a mix of LS-DYNA jobs in an HPC production environment. This development is particularly important in production environments that include other applications and disciplines such as NVH and CFD that might request similar HPC resources as LS-DYNA during a multi-job throughput workload.

Summary and Conclusions

A discussion was provided on the HPC technology requirements of LS-DYNA applications and workflow, including characterizations of the performance behavior typical of LS-DYNA simulations on SGI Altix 3000 systems. Effective implementation of highly parallel LS-DYNA simulations must consider a number of features such as parallel algorithm design, system software performance issues, and hardware communication architectures.

Development of increased parallel capability will continue on both application software and hardware fronts to enable FEA modeling at increasingly higher resolutions. Examples of LS-DYNA simulations demonstrate the possibilities for highly efficient parallel scaling on the SGI Altix servers and cluster systems.

LSTC and SGI will continue to develop software and hardware performance improvements, enhanced features and capabilities, and greater parallel scalability to accelerate the overall solution process of LS-DYNA simulations. This alliance has improved FEA modeling practices in research and industry on a global basis and will continue to provide advancements for a complete range of engineering applications.

References

1. LAUDON, J. and LENOSKI, D., (1997) "The SGI Origin ccNUMA Highly Scalable Server", SGI Published White Paper.

2. WOODACRE, M., ROBB, D. (2003) "The SGI Altix 3000 Global Shared Memory Architecture", SGI Published White Paper.