

14th CIRP Conference on Modeling of Machining Operations (CIRP CMMO)

A High Performance Computing Cloud Computing Environment for Machining Simulations

X. Man^a, S. Usui^a, S. Jayanti^a, L. Teo^a, and T. D. Marusich^{a*}^aThird Wave Systems, 6475 City West Parkway, Minneapolis MN 55344, USA* Corresponding author. Tel.: +1-952-832-5515; fax: +1-952-844-0202. E-mail address: sales@thirdwavesys.com.

Abstract

Machining is a pervasive manufacturing process used in industries such as automotive, aerospace, medical implants and oil and gas. Analysis of processes via physics-based modeling enables new and innovative designs for cutting tools, provides confidence of machined workpiece quality characteristics and allows reduction in machining cycle times and tooling costs. Progressively sophisticated analyses of machining processes have evolved with the inclusion of effects of full three-dimensional analysis of cutting tools and complex tool/workpiece kinematics. Detailed-level analysis of machined workpiece surfaces based on finite element method (FEM) allows prediction of residual stresses, work hardened layer depths and heat flow. However, with the increase in model sophistication has come with computational burden. This paper details a high performance computing (HPC) environment for finite element models used for machining analysis. First, the FEM model is reviewed and its extension to high core-count shared memory environments is described. Scaled performance improvements for up to forty cores are demonstrated and performance improvements documented. Next, an HPC cluster is designed and a specialized batch queuing software is implemented that allows submission, monitoring and management of large scale machining simulations. Finally, an infrastructure for delivering the HPC capability to customers through Software as a Service (SaaS) is introduced.

© 2013 The Authors. Published by Elsevier B.V.

Selection and peer-review under responsibility of The International Scientific Committee of the “14th CIRP Conference on Modeling of Machining Operations” in the person of the Conference Chair Prof. Luca Settineri

Keywords: Modelling; Environment; High performance computing (HPC)

1. Introduction

Machining is a pervasive manufacturing process used in industries such as automotive, aerospace, medical implants and oil and gas. Analysis of processes via physics-based modeling enables new and innovative designs for cutting tools, provides confidence of machined workpiece quality characteristics and allows reduction in machining cycle times and tooling costs. Progressively sophisticated analyses of machining processes have evolved with the inclusion of effects of full three-dimensional analysis of cutting tools and complex tool/workpiece kinematics. Detailed-level analysis of machined workpiece surfaces based on finite element method (FEM) allows prediction of residual stresses, work hardened layer depths and heat flow. However, with the increase in model sophistication has

come with computational burden. This paper details a high performance computing (HPC) environment for finite element models used for machining analysis. First, the FEM model is reviewed and its extension to high core-count shared memory environments is described. Scaled performance improvements for up to forty cores are demonstrated and performance improvements documented. Next, an HPC cluster is designed and a specialized batch queuing software is implemented that allows submission, monitoring and management of large scale machining simulations. Finally, an infrastructure for delivering the HPC capability to customers through Software as a Service (SaaS) is introduced.

2. Finite Element Modeling of Machining Processes

The finite element method used for these machining process analyses is Third Wave AdvantEdge. It is based

on the original work by Marusich and Ortiz [1] and employs an explicit Lagrangian finite element formulation equipped with continuous adaptive meshing technique. The model accounts for dynamics effects, heat conduction, full thermal-mechanical coupling, plasticity in finite deformation and large strain rate, and frictional contact between deformable meshes. A detailed description of the finite element model and its validation can be found in [2]. AdvantEdge's graphical user interface (GUI) enables easy setup of simulations for common machining processes and supports tool geometry import for various CAD formats (e.g. STEP, VRML). The model also provides versatile tools for constitutive modeling, allowing complex material behaviors be adequately described for a wide range of workpiece materials.

2.1. Mechanical and Thermal Computation

The model involves two major computation modules – mechanical and thermal time stepping. Both time stepping algorithms are derived by using finite element method for spatial discretization and explicit time integration schemes for temporal discretization. The mechanical time stepping can be summarized as a central difference integration scheme as follows

$$\begin{aligned} \mathbf{d}_{n+1} &= \mathbf{d}_n + \Delta t \mathbf{v}_n + \frac{1}{2} \Delta t^2 \mathbf{a}_n \\ \mathbf{a}_{n+1} &= \mathbf{M}^{-1} (\mathbf{R}_{n+1}^{ext} - \mathbf{R}_{n+1}^{int}) \\ \mathbf{v}_{n+1} &= \mathbf{v}_n + \frac{1}{2} \Delta t (\mathbf{a}_{n+1} + \mathbf{a}_n) \end{aligned} \quad (1)$$

where \mathbf{R}^{ext} and \mathbf{R}^{int} denote the external and internal force vectors, respectively; \mathbf{M} is the mass matrix; and \mathbf{a} , \mathbf{d} , and \mathbf{v} denote the acceleration, displacement, and velocity vectors, respectively. The subscript $n+1$ indicates that the quantity is associated with the time t_{n+1} , which is advanced from the previous time by $t_{n+1} = t_n + \Delta t$. The thermal time stepping follows the forward Euler scheme and reads

$$\mathbf{T}_{n+1} = \mathbf{T}_n + \Delta t \mathbf{C}^{-1} (\mathbf{Q}_n - \mathbf{K}_n \mathbf{T}_n), \quad (2)$$

where \mathbf{T} denotes the temperature vector, and \mathbf{K} and \mathbf{C} are the conductivity and heat capacity matrixes, respectively. \mathbf{Q} is the heat source vector.

The thermal-mechanical coupling in a typical machining process involves two actions. Heat is generated along the tool-chip interface due to friction and a fraction of the plastic work done in workpiece is also converted to heat. The conversion fraction is usually assumed to be based on Taylor and Quinney's study [3]. The generated heat increases the temperature in the

system and softens the workpiece material. The coupling is modeled by a staggered mechanical-thermal time stepping procedure as outlined by Marusich and Ortiz [1]. A mechanical step is taken first based on the current temperature distribution, and the heat generated is calculated based on the plastic work and friction in the step. Then the temperature distribution is updated based on the new heat source and thermal conduction according to Equation (2). In the next mechanical step, the updated temperature distribution is used as the input to determine the thermal softening and thermal expansion of the materials.

2.2. Contact Model

The impenetrability constraint between tool and workpiece contact regions is enforced by a predictor-corrector scheme developed by Taylor and Flanagan [4]. The algorithm handles contact between deformable meshes. First, the penetration distances for all nodes are calculated based on the predictive configuration, which is obtained by updating the nodal position based on Equation (1). Second, the penetration is eliminated by applying corrective accelerations on the contacting nodes. The correction is consistent with the overall time integration scheme. The friction is modeled by a Coulomb friction model described by Marusich and Ortiz [1].

2.3. Constitutive Model

The constitutive model is based on the stress update method proposed by Cuitiño and Ortiz [5]. The method extends small strain stress update algorithms to finite deformation range at the kinematics level and thus provides a versatile framework for constitutive modeling. The standard constitutive model in Third Wave AdvantEdge assumes the following when defining the flow stress:

$$\sigma(\alpha, \dot{\alpha}, T) = g(\alpha) \Theta(T) \Gamma(\dot{\alpha}), \quad (3)$$

where $g(\alpha)$ and $\Gamma(\dot{\alpha})$ are the isotropic strain hardening and rate sensitivity defined as power law functions, and $\Theta(T)$ the thermal softening function defined as a fifth order polynomial. Flow stress models other than power laws such as Johnson-Cook model in [6] and Zerilli-Armstrong model in [7] can also be implemented and conveniently interfaced with AdvantEdge using a User Defined Yield Surface (UDYS) capability.

2.4. Adaptive Remeshing

Adaptive remeshing is utilized in AdvantEdge to sidestep the difficulty of element distortion inherent in

Lagrangian formulations. The mesh quality is monitored during a simulation, and when element distortion reaches a certain tolerance, adaptive remeshing is triggered. Refinement, improvement, and coarsening meshing operators are applied in various parts of the mesh. Mesh in regions where plastic deformation is active is refined to resolve the large temperature and deformation gradients, and mesh in the regions that have become inactive is coarsened to keep the size of the problem bound. Element distortion is fixed by a combination of mesh refinement and improvement. Certain smoothing is applied to improve the aspect ratios of the elements. Following an adaption, a transfer operator is applied to transfer the nodal and elemental states between meshes. A detailed description of the adaptive remeshing algorithm can be found in [1].

3. Performance Improvement

3.1. Parallel Performance Improvement

The numerics outlined in previous section are computationally intensive. For constitutive modeling, local Newton-Raphson iterations are performed at each integration point at each time step with multiple expensive floating point function calls. The computation is proportional to the size of discretization both in space and in time and is usually the most expensive portion of the code when running in sequential mode. The contact algorithm involves searching mesh entities in space to check potential penetration and thus traverses a large amount of mesh data. The thermal-mechanical coupling and contact correction also entail a considerable amount of floating point operations (FLOPs) and memory access cost. For a machining simulation with refined finite element mesh (>100,000 elements), the solution time running in sequential mode can take days or even weeks depending on the length of cut to be simulated. It is critical to reduce the solution time by improving the performance of the model.

In the past, application developers relied on the increase of clock speed of microprocessors and highly optimized sequential code to achieve performance improvement. Today, as the clock speed of microprocessors plateaus due to heat removal and energy consumption constraints, it becomes critical for an application to fully exploit parallelism in order to achieve performance improvement on the latest multicore and many-core processors, which are designed to yield higher processing capability through multi- and many- way parallelism. Moreover a scalable parallel algorithm design and implementation is the key for an application to continue to enjoy the performance improvement with each new generation of microprocessors.

Motivated by such a vision, this study aims at improving the performance and scalability of the current parallel AdvantEdge FEM model. The parallel programming model is based on shared memory systems and the code has been parallelized with OpenMP in some computation intensive modules in its earlier versions. However, in AdvantEdge 6.0, major changes in both algorithms and implementation are made to improve the strong scaling performance of the code on the latest Intel Xeon multicore processors.

The constitutive modeling and internal force computation is the most expensive part of the code. Two looping structures are involved in this task. On the element level, the constitutive update and elemental internal force can be safely parallelized as the computation at each integration point is independent of others. Since this task is usually computation bound, it scales strongly as the number of parallel cores increases. On the node level, global forces are assembled based on the local elemental forces. For unstructured finite element mesh, since there does not exist such a data structure as a stencil in finite difference models, which can explicitly define the coupling between elements and/or nodes, synchronization is needed when the global vector is constructed. A special algorithm is designed to minimize the synchronization cost and to maximize the parallelism in the module. The algorithm is also used in contact correction, where a similar two-level looping structure exists to apply the local and global corrections.

According to Amdahl's law, the overall performance of a parallel code is bound by the sequential portions of the code. To improve the strong scaling of the entire code, other less scalable portions of the code are identified and improved as well. The nodal kinematics update is completely parallelized in AdvantEdge 6.0 and the sequential part is removed. Thermal computation is less expensive as far as the floating point operations (FLOPs) are concerned. However, the data transfer for the thermal-mechanical coupling is found to be a performance bottleneck due to memory access. The implementation is optimized by coupling the two in a tighter manner such that the data exchange becomes more cache friendly. This change effectively removes the memory access bottleneck and significantly reduces the wall clock time for the operation. A new contact algorithm is designed to exploit more parallelism and to reduce synchronization and it improves the scalability of the contact model for larger contact problems.

3.2. Parallel Benchmarking Results

The aforementioned parallel performance improvements are benchmarked using three simulations: an indexable milling of Ti-6Al-4V, a solid endmilling of Al7050, and a solid drilling of Ti-6Al-6V. Since these

cases cover typical material, tooling, and process types, the benchmarking battery gives a fairly good representation of different types of AdvantEdge simulations. Two types of hardware are used in the benchmarking: one has two 6-core Intel Xeon X5680 processors at a clock speed of 3.33 GHz with 12 MB L3, and the other has four 10-core Intel Xeon E7-4860 processors at a clock speed of 2.4 GHz with 30 MB L3.

The performance is measured by the total elapsed time for a simulation to finish and the baseline performance is established by running the benchmarking cases with AdvantEdge 5.9. Performance improvement achieved in AdvantEdge 6.0 is studied by comparing the total elapsed time of version 6.0 simulations with version 5.9 (baseline). Parallel scalability is studied by running the benchmark cases with different numbers of parallel cores and then calculating the speedup $s = T_1/T_n$, where n denotes the number of cores used, and T_1 and T_n the elapsed time using one core and n cores, respectively. Since AdvantEdge 5.9 can only support up to eight cores, the baseline simulations are run with one, two, and eight cores on both benchmarking computers. For AdvantEdge 6.0, benchmarking on the 12-core computer is done for one, two, eight, and eleven cores; and one, two, eight, sixteen, and 38 cores on the 40-core computer.

Figure 1 shows the total elapsed time for the benchmarking cases on the 12-core computer and significant reduction in total elapsed time is observed with version 6.0. Stronger scalability for high core counts is achieved in version 6.0, as shown in Figure 2. Of these three benchmarking cases, the most expensive simulation (drilling) shows the strongest scalability and largest performance improvement. Figure 3 and Figure 4 show the 40-core benchmarking data.

To ensure the correctness of the parallel implementation, simulation results obtained in AdvantEdge 6.0 with parallel performance enhancements are compared against the baseline results obtained with a single core. Figure 5 shows for the drilling case the temperature contour of AdvantEdge 5.9 running with a single core and AdvantEdge 6.0 running 38 cores. Figure 6 shows the torque comparison.

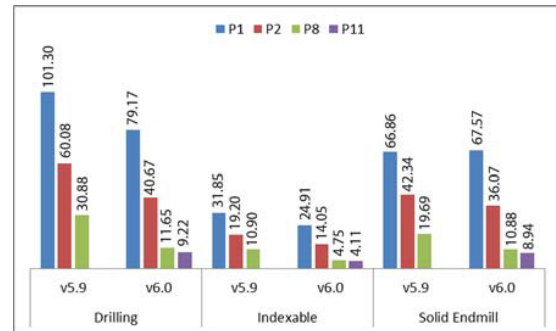


Figure 1: Total elapsed time (h) on 12-core.

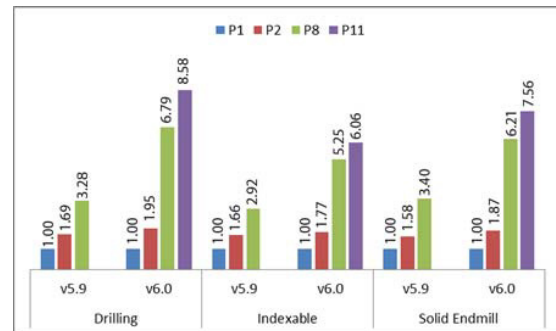


Figure 2: Parallel speedup on 12-core.

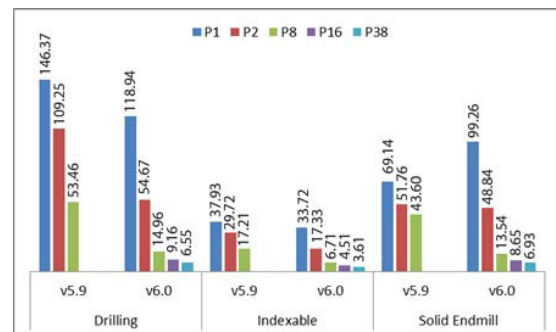


Figure 3: Total elapsed time (h) on 40-core.

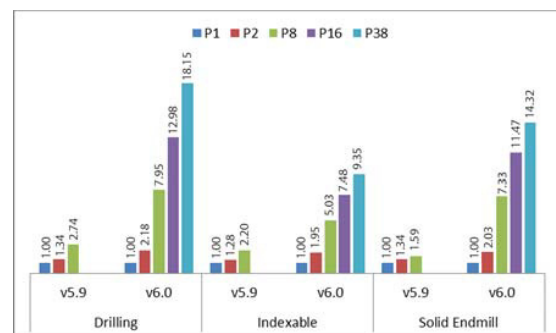


Figure 4: Parallel speedup on 40-core

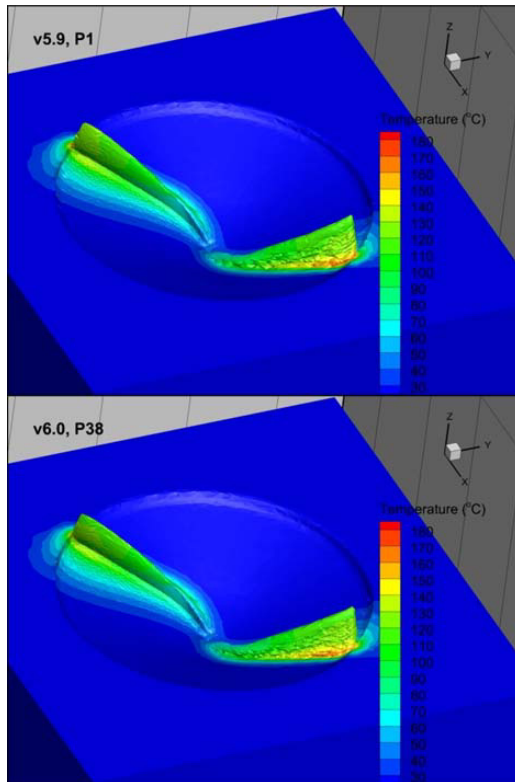


Figure 5: Temperature contour comparison for the drilling case.

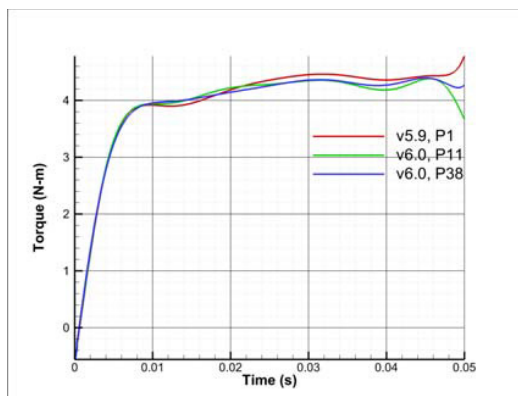


Figure 6: Torque comparison for the drilling case.

4. HPC Cluster

The parallel performance improvement reduces the solution time for a single simulation. However, within Third Wave Systems (TWS), hundreds of machining simulations are usually running concurrently for various development and application activities. How to efficiently use the available computing resources and improve the performance of the entire system becomes a challenge. An HPC system has been built to provide the computing capacity and to manage the computing

resources efficiently. This HPC system is developed based on the Windows HPC technology. Figure 7 shows the schematics of the system, which consists of an HPC Client, HPC Server, and cluster of compute nodes.

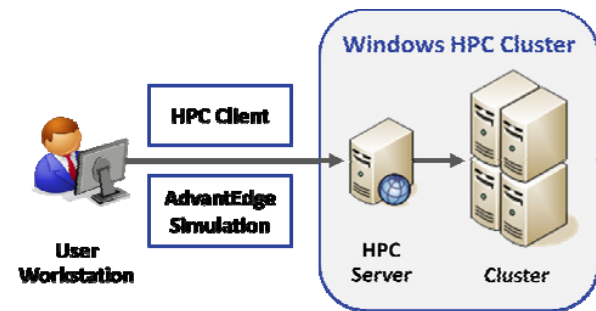


Figure 7: TWS HPC system schematics.

The HPC Client interface allows users to submit AdvantEdge simulations as jobs to the HPC Server, while the HPC Server manages a job queue and dispatches a job when resources become available on the computing cluster. Multiple clients can simultaneously submit jobs and job statuses are continuously updated. Once a simulation starts running, the user can monitor its progress and even cancel it through the HPC Client. Figure 8 shows a snapshot of the HPC Client interface.

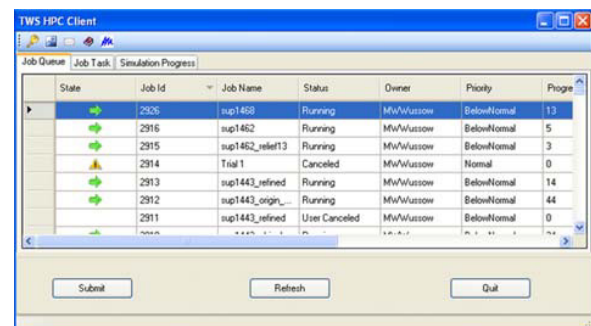


Figure 8: TWS HPC Client interface

The HPC Server runs on a Dell PowerEdge R720xd head node and Windows Server 2012 Standard Edition. The HPC cluster consists of 23 12-core compute nodes using Intel Xeon X5680 processors and four 40-core compute nodes using Intel Xeon E7-4870 processors. The HPC cluster and queuing system organizes the computing resources in a scalable way and enables their full utilization.

5. Software as a Service (SaaS)

With the development of cloud computing, TWS is building the infrastructure for delivering AdvantEdge

Software as a Service (SaaS). In the SaaS model (Figure 9), AdvantEdge is hosted in the TWS HPC environment while customers remotely upload and submit AdvantEdge simulations via the Internet. SaaS Client resides locally on a customer workstation and it communicates with TWS SaaS Web Server for simulation submission and status update. SaaS Web Server populates a SaaS Job Database, which keeps track of user accounts, company affiliations and simulation status, and the Web Server also allows users to track their simulations via querying into the aforementioned database. File Manager handles the file transfer between SaaS Client and TWS, which includes simulation input file upload and result file download. SaaS Job Manager is the interface between the SaaS Job Database and the HPC Cluster. It queries the database for each job's status and assigns available computing resources on the HPC Cluster. If a job is in queue and resources become available, Job Manager will gather all the corresponding input files of the job through File Manager and submit it to the cluster. Once a job is submitted, its status is changed to running mode and will be updated by the Job Manager based on its status on the HPC cluster. That status is relayed by the Web Server to the Client so that the customer can monitor the progress of the simulation.

Security of the entire infrastructure will be ensured by transferring all data via a secure pipeline enabled by the secure socket layer (SSL) encryption mechanism. Customer authentication and authorization will be managed via up-to-date customer account databases managed by TWS sales and support engineers.

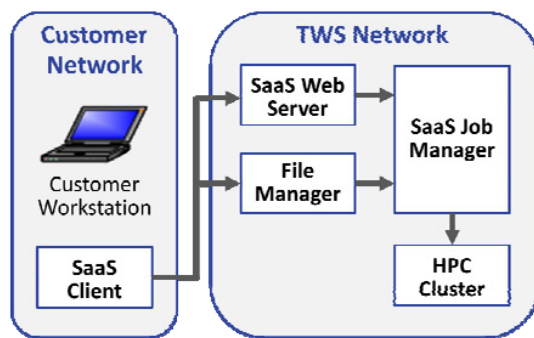


Figure 9: TWS AdvantEdge SaaS model.

The SaaS model delivers to customers the state-of-the-art computing environment for machining simulations and enables customers to focus on their problem solving with lower cost on computing hardware and software investment and maintenance.

6. Conclusion

In this paper, a high performance computing environment for machining simulations is introduced. It is comprised of a finite element analysis software package with demonstrated strong scalability using the latest multicore technology, a scalable HPC cluster with the state-of-the-art computing hardware, and a flexible yet secure SaaS delivery model. This system delivers the powerful machining analysis tool to cutting tool designers and manufacturers and enables them to solve their analysis problems with less time and lower cost.

References

- [1] Marusich, T. D. and Ortiz, M., 1995, Modelling and Simulation of High-Speed Machining, *Int. J. Num. Meth. Eng.* 38:3675-3694.
- [2] Man, X., Ren, D., Usui, S., Johnson, C., Marusich, T. D., 2012, Validation of Finite Element Cutting Force Prediction for End Milling, *Procedia CIRP* 1:663-668.
- [3] Taylor, G. I. and Quinney, H., 1931, The Plastic Distortion of Metals, *Philos. Trans. Roy. Soc. London*, A230:323-362.
- [4] Taylor, L. M. and Flanagan, D. P., 1987, PRONTO 2D: A Two-Dimensional Transient Solid Dynamics Program, SAND86-0594.
- [5] Cuitiño, A. and Ortiz, M., 1992, A Material-Independent Method for Extending Stress Update Algorithms from Small-Strain Plasticity to Finite Plasticity with Multiplicative Kinematics, *Engineering Computations*, 9:437-451.
- [6] Johnson, G. R. and Cook, W. H., 1983, A Constitutive Model and Data for Metals Subjected to Large Strains, High Strain Rates and High Temperatures, *Proceedings of the 7th International Symposium on Ballistics*, 541-546.
- [7] Zerilli, F. J. and Armstrong, R. W., 1987, Dislocation-Mechanics-Based Constitutive Relations for Material Dynamics Calculations, *Journal of Applied Physics*, 61:1816-1825.