The Teraflop Workbench — Enhancing High Sustained Performance on Vector Systems

Sabine ROLLER^{1*}, Michael RESCH¹, Martin GALLE², and Wolfgang BEZ²

¹*High Performance Computing Center Stuttgart (HLRS)*

²*NEC High Performance Computing Europe GmbH*

Received May 5, 2008; final version accepted January 7, 2009

This paper provides a comprehensive collection of applications from different fields that have all one thing in common: They are typical representatives of their research field with high need for High Performance Computing resources, and they all show high sustained performance on the HPC vector system NEC-SX8. The paper also describes future needs to address when describing the next generation of applications.

KEYWORDS: High Performance Computing (HPC) Applications, Teraflop Workbench, Multi-Physics/Multi-Scale Simulations

1. Introduction

The Teraflop Workbench is a joint project by HLRS and NEC HPCE. Its mission is to support users on HPC systems in their ambition to efficiently run their applications on the available hardware resources in the shortest possible turnaround times. The basic idea is to bring together application knowhow, physical background, numerical mathematics, computer science, and computer engineering to enable leading edge computational science. The applications presented in this paper now all deliver TFLOP/s sustained performance on the NEC-SX8 system providing the opportunity to solve larger problems and to gain new insight into different areas of research. Application areas addressed are from automotive engineering, aerospace engineering, civil engineering, combustion, turbo machinery, molecular dynamics, astrophysics, medical surgery, bioinformatics, and more.

2. HPC Installation NEC-SX8 at HLRS

In this section, an overview of the NEC SX-8 installation at the HLRS is given including detailed technical descriptions of the most important modules. Figure 1 illustrates the NEC installation at HLRS. The main building block is the 72 nodes of the NEC SX-8. The SX-8 nodes are connected with the Internode Crossbar Switch (IXS). NEC's global file system gStorageFS enables the entire multi-node complex to view a single coherent file system. Two NEC TX-7 Itanium servers are employed as front-end servers to the SX-8 nodes. Each TX-7 system is equipped with 32 Itanium II 1.5 GHz CPUs and a large main memory of 256 GB and 512 GB, respectively. Additionally, 200 PC cluster nodes with two Intel Xeon EM64T (Nocona) 3.2 GHz CPUs are available. They are connected with an Infiniband high speed network. The idea behind this installation is to provide an integrated workbench for preprocessing, simulation, postprocessing and visualization where it is possible to choose the best platform for each task.

3. Achievements in Today's Applications — Some Selected Examples

Within the framework of the Teraflop Workbench more than 10 applications from different scientific areas accomplished a performance beyond 1 TFLOP/s. The application performance delivered by the new system is two orders of magnitude higher than the predecessors SX-4/32 and SX-5e/16. This is achieved by optimizing the CPU performance as well as the scaling behavior. Furthermore, improvements of the complete process workflow due to the integration of different modules into a “hybrid vector system” was carried out. This enables our researchers to gain much more detailed insight in the simulated entities and to enter new scientific fields.

The aim of the Teraflop Workbench project is to support users in efficiently using the available hardware resources installed at HLRS. It is not the aim of the project to provide a benchmark suite. Thus, the selection of the applications is not representative in that sense, but covers topics important to the users of HLRS' vector system.

* **Contact address:** HLRS, Nobelstr. 19, 70569 Stuttgart, Germany

Fig. 1. Installation at HLRS.

3.1 Success Stories I: N3D

The performance of sub-, trans- and supersonic fluid-dynamic devices may be strongly affected by laminar-turbulent transition and unsteady flow phenomena. Due to their complexity, most of these problems are not yet fully understood. Resolving all relevant flow structures in space and time, Direct Numerical Simulations (DNS) provide a detailed insight into the underlying physical processes.

N3D is a solver for highly accurate computation of transitional and turbulent flows, developed at the Institute for Aerodynamics and Gasdynamics (IAG), Stuttgart of University. It is based on the vorticity-velocity formulation of the Navier-Stokes equations for an incompressible fluid. The code is optimized for highest processing speed on each single vector-type CPU. To obtain optimal performance on multinode vector supercomputers, N3D applies a hybrid approach employing two fundamentally different parallelization techniques at the same time: a directive- based shared-memory OpenMP parallelization is combined within one node with an explicitly coded distributed-memory MPI parallelization. The performance on vector supercomputers can be about 50–60% of the theoretical peak performance. The code scales very well in distributed-/shared-memory environments with a high memory bandwidth such as the NEC SX series.

Using the improved N3D program on the SX-8, the researchers of IAG were able to make quite large calculations for laminar turbulent transition at wings of commercial aircrafts:

- They enlarged the calculation area from a 2 cm to a 30 cm wide part
- They went from artificial flow conditions to real flight flow conditions from measurements

With this, the researchers have been able to develop a new approach for turbulence reduction and confirm its general validity. The new method is expected to reduce fuel consumption by about 15%. A patent for this new method is pending.

3.2 $\mathcal{BES}\mathcal{T}$: Lattice Boltzmann Flow Solver

The flow solver $\mathcal{BES}\mathcal{T}$ was developed at the Lehrstuhl für Strömungsmechanik (LSTM), Universität Erlangen-Nürnberg. It utilizes the lattice Boltzmann method to solve two or three dimensional incompressible flow problems. This method solves a discretized kinetic equation from which the equivalent hydrodynamic fields can be later extracted, instead of directly solving the Navier-Stokes together with the continuity equation Chen and Doolen (1998). The code is applied in various scientific and engineering disciplines such as turbulent flows and aerodynamics, chemical and process engineering and bio and environmental processes. Figure 2 illustrates two typical applications. The left figure is an example for flow through a highly porous media Zeiser *et al.* (2002) and the picture on the right shows a direct numerical simulation of wall bounded turbulence Lammers (2005). The SX-8 is mainly used for compute intensive direct numerical simulations.

Fig. 2. Lattice Boltzmann simulation of a flow through a highly porous media.

Fig. 3. Maximum sustained performance achievable on HLRS' NEC SX-8 system.

To make optimal use of the SX-8's capabilities numerical grids with more than a million points are necessary. Therefore a typical application would be direct numerical simulations for example. For the mentioned grids, the achievable efficiency is about 75% of the vector peak performance on a single CPU and still more than 60% on the whole installation. Thus, the maximum achievable sustained performance is 5.7 TFLOP/s on the whole machine.

In Fig. 3 scaling behavior for typical production grids of a turbulent plane channel are plotted. The strong scaling results for a grid with 126 million points is given in the left plot. On one node $\mathcal{BET}$ reaches 85 GFLOP/s. The decline in the performance is roughly 10% when using 16 nodes instead of one. The strong scaling grid in the right plot consist of $444 \times$ million points. From 8 to 64 nodes are used for the benchmark. Additionally, in the right plot weak scaling results with 7 million points per CPU are plotted. On 64 nodes a performance of 3.3 TFLOP/s is reached for the strong scaling case and 4.4 TFLOP/s if the domain size per CPU is fixed. The single CPU efficiency is 75% of the peak performance. On the complete HLRS installation 5.7 TFLOP/s can be achieved (see Fig. 3). For performance comparison with other HPC systems the reader is referred to Wellein *et al.* (2006) and Lammers and Küster (2006).

3.3 Linear Equation Solver Development

Many scientific problems require the solution of large and sparse linear systems of equations. Unfortunately, available packages like PETSc, Aztec, or Trilinos are not optimized for Vector Parallel architectures. The development of a respective highly optimized and general toolbox is a central activity within the Teraflop Workbench. The outcome of this activity will be used for a variety of applications and lift them to higher performance.

Problems with public domain sparse solvers on vector systems are mainly due to

- Short vector length (sparse storage formats)
Most solvers use natural row-oriented sparse storage formats where as Jagged Diagonal format is needed on vector machines
- Indirect memory addressing (variables at each grid point are typically not blocked)
Block-based computations necessary
- Factorization based DD preconditioning (heavily recursive)
ILU, ICC on subdomains
- Vectorization
Coloring needed for Gauss-Seidel, incomplete factorization, ...
- Excessive function calls are common with Object Oriented approach

The single CPU performance improvement in sparse matrix-vector product, a key kernel in any sparse iterative solver is shown in Fig. 4. With real applications, the sustained per processor performance using BLIS reaches five Gflop/s in the whole solver on the NEC SX-8 which is about 30% peak performance of the CPU Present focus is on implementation of robust scalable preconditioners for the Krylov subspace methods such as the algebraic multigrid method.

4. Future Applications

Another important goal of the Teraflop Workbench is to demonstrate how Linux clusters and SMP systems can be integrated with a vector system to ensure an excellent pre-processing–simulation–post-processing–visualization workflow. The point is that each of the three steps contains several sub-tasks which are typically not suited for the same hardware architecture. While in the past, at least each step in itself could efficiently use one architecture only, this is

Fig. 4. Single CPU performance of Sparse MVP on NEC SX-8.

currently changing. To show examples, we concentrate in this paper on the simulation step. Here, we see applications consisting of different modules which have to be run on different architectures to efficiently use the hardware. We find parts vectorizing very well and others that don't. Therefore we investigate the coupling of codes and architectures in a hybrid sense.

4.1 Coupled Applications—Coupled Systems

The perspective of future applications is the observation that in several scientific numerical experiments, the step closer to reality is no longer achieved by higher resolution only, but also by more accurate modeling. This might be

- include more physical phenomena within one code (more equations, e.g. chemical reactions in fluid flow)
- by coupling independent modules (e.g. fluid-structure interaction)
- by running different simulations one after the other (workflow, e.g. medical surgery, interactive visualization)

Examples come from different research fields. The most mature are fluid-structure interactions as performed by the Institute for Hydraulic Machinery (IHS) of the University of Stuttgart. The costly flow simulation is performed with the IHS code FENFLOSS on SX-8, showing high performance of 2.6 TFlop/s on the vector system. It is coupled to the structure simulation performed with the commercial code Abaqus. For Abaqus, a small PC cluster is sufficient and also necessary since the usage is restricted by the availability of proprietary licences. The coupling code MpCCI, developed by Fraunhofer Gesellschaft mbH can be also run on a small PC cluster or workstation computer. The whole workflow including preprocessing, steering and simulation is provided and controlled in the collaborative Environment COVISE in a cave on a visualization cluster. Advantages of the usage of different simulation codes are:

- Separate modeling
- Established codes
- No changes in codes
- Data transfer via interpolation
- Distinct resource requirements
- Different availability

The disadvantage on the other hand is, that things are getting more and more complex. People are not experts in every detail. This leads to new requirements on HPC Services. User want to be able to use some codes as black box, they need tools (e.g. MpCCI to couple 2 codes without changing the codes). Users need support in interdisciplinary work (e.g. workflow tools/portals/...) such that they can concentrate on their own purposes. From system point of view, co- or conditional scheduling tools are needed. For stronger coupled problems, it is necessary to provide a cross-MPI that enables processes to run on different parts of the heterogeneous system setup while for the user, a single view on his application is maintained.

5. Future Requirements—Mapping of Applications to Architectures

Upcoming applications as described above often require different hardware for different parts of the simulation run. On the HPC architecture, this leads to an emphasize on hybrid Systems. The point behind the scenes are on the one hand the distinct requirements of the various parts. On the other hand, distributed systems are needed since data and/or codes are not available everywhere (e.g. licences of ISV codes or existence of executables only for standard Linux distributions), or sometimes because data are not allowed to be shared (e.g. medical data). Currently, two ways of developments can be observed: As integrated systems on the one hand, as systems tightly connected to standard platforms.

From software point of view, global file systems, coupled MPI libraries, scheduler with advanced functionalities are necessary as well as hybrid debugger and performance analysis tools. Workflow and meta-languages for coupling of applications that are under development in the Grid computing community might be helpful in future HPC applications as well.

6. Summary and Conclusions

The presented applications demonstrate that the NEC SX-8 is a very powerful tool for computational science in different application fields. The achievable application performance is between 30 and 60% of peak, depending on the problem size and the number of processors used. Additionally, it has been shown that the SX-8 is not only able to compute large cases with high performance and high efficiency, but also medium sized cases run with little performance degradation on a large number of processors. Therefore, the system allows on one hand the solution of new and larger problems leading to new insights. On the other hand, it offers the opportunity to shorten turnaround times and to speed up development cycles with high cost efficiency.

REFERENCES

- [1] Chen, S., and Doolen, G. D., “Lattice Boltzmann Method for Fluid Flows,” *Annu. Rev. Fluid Mech.*, **30**: 329–364 (1998).
- [2] Lammers, P., Direkte numerische Simulationen wandgebundener Strömungen kleiner Reynoldszahlen mit dem lattice Boltzmann Verfahren, PhD thesis, Universität Erlangen-Nürnberg, Germany (2005).
- [3] Lammers, P., and Küster, U., Recent Performance Results of the Lattice Boltzmann Method, In inSiDE, Innovatives Supercomputing in Deutschland, Vol. 4. HLRS, LRZ and NIC, <http://inside.hlr.de/> (spring 2006).
- [4] Wellein, G., Zeiser, T., Hager, G., and Donath, S., “On the Single Processor Performance of Simple Lattice Boltzmann Kernelss,” *Computers & Fluids*, **35**: 910–919 (2006).
- [5] Zeiser, T., Steven, M., Freund, H., Lammers, P., Brenner, G., Durst, F., and Bernsdorf, J., “Analysis of the flow field and pressure drop in fixed bed reactors with the help of lattice Boltzmann simulations,” *Philos. Trans. R. Soc. London, Ser. A*, **360** (1792): 507–520 (2002).