High Performance Computing on Vector Systems 2008

Michael Resch · Sabine Roller · Katharina Benkert ·
Martin Galle · Wolfgang Bez · Hiroaki Kobayashi ·
Toshio Hirayama
Editors

High Performance Computing on Vector Systems 2008

Springer

Michael Resch
Sabine Roller
Katharina Benkert

Höchstleistungsrechenzentrum
Stuttgart (HLRS)
Universität Stuttgart
Nobelstraße 19
70569 Stuttgart
Germany
resch@hlrs.de
roller@hlrs.de
benkert@hlrs.de

Martin Galle
Wolfgang Bez

NEC Deutschland GmbH
Hansaallee 101
40549 Düsseldorf
Germany
mgalle@hpce.nec.com
wbez@hpce.nec.com

Hiroaki Kobayashi

Cyberscience Center
Tohoku University
6-3 Aramaki-Aza-Aoba
Sendai, 980-8578
Japan
koba@isc.tohoku.ac.jp

Toshio Hirayama

Center for computational science and e-systems
Japan Atomic Energy Agency
Sumitomo fudosan Ueno Bldg. No. 8
6-9-3 Higashi-Ueno Taito-ku
Tokyo, 110-0015
Japan
hirayama.toshio@jaea.go.jp

Front cover figure: Simulation of the UV curing process in automotive coating with multiple ultraviolet lamps. Picture due to IFF, University of Stuttgart, Germany and BMW Group.

ISBN 978-3-540-85868-3 e-ISBN 978-3-540-85869-0

DOI 10.1007/978-3-540-85869-0

Library of Congress Control Number: 2008934396

Mathematics Subject Classification (2000): 68Wxx, 68W10 , 68U20, 76-XX, 86A05, 86A10, 70Fxx

Cover design: WMXDesign, Heidelberg

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

This book covers the results obtained in the Teraflop Workbench project during a four years period from 2004 to 2008. The Teraflop Workbench project is a collaboration between the High Performance Computing Center Stuttgart (HLRS) and NEC Deutschland GmbH (NEC-HPCE) to support users to achieve their research goals using high performance computing.

The Teraflop Workbench supports users of the HLRS systems to enable and facilitate leading edge scientific research. This is achieved by optimizing their codes and improving the process workflow which results from the integration of different modules into a “hybrid vector system”. The assessment and demonstration of industrial relevance is another goal of the cooperation.

The Teraflop Workbench project consists of numerous individual codes, grouped together by application area and developed and maintained by researchers or commercial organizations. Within the project, several of the codes have shown the ability to reach beyond the TFlop/s threshold of sustained performance. This created the possibility for new science and a deeper understanding of the underlying physics. The papers in this book demonstrate the value of the project for different scientific areas.

The work in the Teraflop Workbench project gave us insight into the applications and requirements for current and future HPC systems. We observed the emergence of multi-scale and multi-physics applications, the increase in interdisciplinary work and the growing tendency to use today’s stand-alone application codes as modules in prospective, more complex coupled simulations. At the same time, we noticed the current lack of support for those applications. Our goal is to offer an environment to our users that allows them to concentrate on their area of expertise without spending too much time on computer science itself.

We would like to thank all the contributors of this book and of the Teraflop Workbench project in general.

Stuttgart, July 2008

Sabine P. Roller
Michael M. Resch

Contents

I Future Architectures

First Experiences with NEC SX-9 3
Hiroaki Kobayashi, Ryusuke Egawa, Hiroyuki Takizawa, Koki Okabe, Akihiko Musa, Takashi Soga, and Yoichi Shimomura
1 Introduction 3
2 System Overview 5
3 Performance Evaluation 6
4 Conclusions 9
References 10

Scalable Computing in a Hybrid System Architecture 13
Wilfried Oed
1 Introduction 13
2 The Need for Scalability 14
3 Processor Trends 15
4 The Cray $XT5_h$ Hybrid Supercomputer 17
5 The Cray Cascade Program 19
6 Conclusion 20
References 20

Prerequisites for the Productive Usage of Hybrid Systems 23
Danny Sternkopf and Martin Galle
1 Introduction 23
2 HLRS System Overview 24
3 SW Components for a Hybrid System 25
  3.1 Batch System 26
  3.2 Global Filesystem 27
  3.3 MPI 30
4 Conclusion 31
References 31

II Multiscale and Multiphysics Simulations

Complex Numerical Simulations in Production Techniques 35
A. Scheibe and E. Westkämper
1 Problem Statement and Motivation 35
2 Multi-Scale Modeling and Simulation of Manufacturing Systems . 36
2.1 Vision and Scientific Impact 36
2.2 Challenges, Theoretical Considerations and Definitions .. 37
3 A New Concept Towards a Comprehensive and Holistic Factory Simulation Model 38
3.1 Theoretical Aspects 38
3.2 Vectorial Modelling and Simulation Approach Based on Business/Organizational Parameters and Technical Tolerances 40
4 Complex Example in Automotive Coating Industry 42
5 Conclusions and Future Work 44
References 44

Multi-scale and Multi-physics Applications — User Requirements for Future Applications 45
Sabine P. Roller
1 Introduction 45
2 A Glance at Past, Current and Future Simulation Applications 46
3 Future Application Perspective 47
4 Examples 48
4.1 Dynamic Coating Simulation 48
4.2 Ocean Water Plant 48
4.3 Particle-loaden Flows 49
4.4 Multi-scale Aero-acoustics 50
5 Future User Perspective 51
6 Requirements, Architecture and Tools 52
References 52

III Grid Computing & Data Analysis

An Application of the NAREGI Grid Middleware to a Nationwide Joint-Use Environment for Computing 55
Eisaku Sakane, Manabu Higashida, and Shinji Shimojo
1 Introduction 55
2 NAREGI Grid Middleware 56
3 Issues 57
3.1 Resource Provisioning 58
3.2 Password / Passphrase Management 58
3.3 Issuance of Grid Certificate 58
3.4 Grid Operations 59
4 Application Method 59

5 Evaluation of Proposed Method 61
6 Cooperative Evaluation 62
7 Summary and future works 63
References 63

Interoperation between Atomic Energy Grid Infrastructure (AEGIS) and Other Grids 65
Yoshio Suzuki, Noriyuki Kushida, Naoya Teshima, Kohei Nakajima, Akemi Nishida, and Norihiro Nakajima
1 Introduction 65
2 History of Grid Computing Technology at CCSE 66
3 Atomic Energy Grid Infrastructure 68
4 Interoperability Technology based on Atomic Energy Grid Infrastructure 69
5 Collaboration with the High Performance Computing Center Stuttgart (HLRS) 72
6 Collaboration with Grid-TLSE Project Partners 74
7 Collaboration with the U.S. Department of Energy Under the Global Nuclear Energy Partnership 74
8 Summary 75
References 76

Parallel File Systems in European Grid Projects 79
Peter W. Haas and Michael M. Resch
1 Introduction 79
2 The HLRS Framework 80
  2.1 Target 80
  2.2 Organization 80
  2.3 HLRS Computer Configuration 80
3 Ex@Grid Framework 81
  3.1 Gauss Center for Supercomputing 81
  3.2 Ex@Grid Design 81
4 Consolidation in the Datacenter 82
  4.1 Datacenter Ethernet 82
  4.2 IEEE 802.3ar: Congestion Management 82
5 Multicluster Parallel File Systems 83
  5.1 Multicluster GPFS in DEISA 83
  5.2 Network-centered parallel HSM systems 83
  5.3 Projected mass storage system at HLRS 84
6 The Teraflop Workbench Concept 84
  6.1 Workflow Example 85
  6.2 Fenfloss 85
  6.3 Demonstration in the HLRS Cave 85
  6.4 Virtual Tour of Kiebingen Water Power Plant 86
7 Conclusion 86
References 87

Development of Cognitive Methodology based Data Analysis System 89
Chiaki Kino, Yoshio Suzuki, Noriyuki Kushida, Akemi Nishida, Sachiko Hayashi, and Norihiro Nakajima
1 Introduction 89
2 The Basic Idea of CDAS 91
2.1 Data Analysis Process 91
2.2 VV and DD Functions 91
2.3 Synthesis Function 92
2.4 Evaluation and Judgment 92
3 System Configuration 93
3.1 Flow of Data Analysis by CDAS 93
3.2 Implementation on a Grid Computing 94
4 Result and Discussion 94
5 Conclusions and Future Works 96
References 97

IV Chemical Applications

3D-Flame Modelling in Power Plant Applications 101
Benedetto Risio, Norbert Paßmann, Friedhelm Wessel, Egbert Reinartz
1 Introduction 101
2 Flame modelling tool and computer hardware 102
3 Neurath A/B boiler model 104
4 Simulation results 105
5 Summary 110
References 110

Hierarchical Modeling of Combustion Processes 111
Ulrich Maas, Viatcheslav Bykov, Andriy Rybakov, and Rainer Stauch
1 Introduction 111
2 Dynamics of Reacting Flows 112
3 Detailed Models 114
3.1 Chemical kinetics 115
3.2 Chemistry-Turbulence Coupling 115
3.3 Modeling of Multi-Phase Processes 116
4 Model Reduction 118
4.1 Chemical Kinetics 119
4.2 Chemistry-Turbulence Interaction 121
4.3 Multi-Phase Processes 123
5 Coupling of the Sub-Models 123
6 Summary 125
References 126

Understanding Molecular Recognition and Self-Assembly from Large-Scale Numerical Simulations 129
Stephan Blankenburg and Wolf Gero Schmidt
  1 Introduction 129
  2 Computational Method 131
  3 Results and Discussion 131
  4 Conclusions 136
  References 137

Large Scale Particle-in-cell Plasma Simulation 139
Seiji Ishiguro
  1 Introduction 139
  2 Parallelization of 3D Particle-in-cell Code using High Performance Fortran 140
  3 PIC Simulation of Blob Transport 141
  4 Conclusion 143
  References 144

Multi-scale Modeling of Crack Propagation 145
Mitsuhiro Itakura, Hideo Kaburaki, Masatake Yamaguchi, and Ken-ichi Ebihara
  1 Introduction 145
  2 Multi-scale Model of SCC 146
    2.1 Oxygen Embrittlement Mechanism 146
    2.2 Mechanics Modeling of Crack Growth 147
    2.3 Oxygen Diffusion Modeling 149
  3 Simulation Details and Results 149
  4 Conclusions 151
  References 152

V Climate Modeling, Hydro- and Aerodynamics

The Climate Model ECHAM5 on NEC SX-8 155
Stefan Borowski
  1 Introduction 155
  2 Runtime Analysis 156
  3 Optimizations 157
    3.1 Single CPU Optimizations 157
    3.2 Scalability Optimizations 158
  4 Performance Results 158
    4.1 Single CPU Performance 158
    4.2 Scalability 159
  5 Conclusions 161
  References 162

A Large Spectrum of Free Oceanic Oscillations 163
Malte Müller
1 Introduction 163
2 State of Knowledge 164
3 Free Oceanic Oscillation Model with Consideration of LSA 165
3.1 Theory 166
3.2 The Implicitly Restarted Arnoldi Method 167
3.3 The Parallelization with MPI 169
3.4 The Performance of the Model 169
4 Results 170
4.1 Gravitational Modes 171
4.2 Vorticity Modes 171
5 Conclusion 174
References 174

Direct Numerical Simulation of Controlled Shear Flows 177
Markus J. Kloker, Tillmann A. Friederich, Jens Linn
1 Laminar-Flow-Control Case 177
1.1 Introduction 177
1.2 Numerical Model 179
1.3 Secondary Instability and Control Setup 180
1.4 Control Results and Conclusions 181
2 Effusion Cooling Case at <Mach 6.8> Flight Conditions 184
2.1 Introduction 184
2.2 Numerical Model 184
2.3 Blowing Through Slits 185
2.4 Blowing Through Holes 188
3 Computational Aspects and Outlook 191
References 193

Fluid-Structure Interaction in Turbine Simulation 195
Felix Lippold and Albert Ruprecht
1 Introduction 195
2 Basic Equations 196
3 Fluid-Structure Coupling 200
4 Efficient Moving Mesh Scheme 201
5 Coupled Codes and Architecture 202
5.1 FENFLOSS — CFD on Vector Systems 202
5.2 FSI Coupling with FENFLOSS on SX-8 203
6 von Karman Vortex Street 204
7 FSI Benchmark Application 206
8 Tidal Turbine Runner 209
8.1 Computational Model and Performance 210
8.2 Fluid-Structure Interaction Results 210
9 Summary 212
References 213

Heterogeneous Parallel Aero-Acoustics Using PACX-MPI 215
Harald Klimach and Sabine Roller and Claus-Dieter Munz
1 The Aero-Acoustic Application 215
1.1 Structure of the Application 216
1.2 Coupling Scheme 216
1.3 Parallelization 217
2 PACX-MPI 218
2.1 Communication Layout 218
2.2 Heterogeneous Environment 219
2.3 Starting an Application Using PACX-MPI 219
3 Heterogeneous Computations 219
4 Conclusion 221
References 222

Meandering of Wing-Tip Vortices Interacting with a Cold Jet in the Extended Wake 223
Frank T. Zurheide, Matthias Meinke, and Wolfgang Schröder
1 Introduction 223
2 Governing Equations 225
3 Boundary Conditions 226
3.1 Experimental set-up 226
3.2 Inflow boundary conditions 227
4 Results 228
5 Computational Resources 238
6 Conclusions 239
References 240

I

Future Architectures

First Experiences with NEC SX-9

Hiroaki Kobayashi, Ryusuke Egawa, Hiroyuki Takizawa, Koki Okabe, Akihiko Musa, Takashi Soga, and Yoichi Shimomura

Abstract This paper presents the new supercomputer system NEC SX-9 that has been installed at Tohoku University in March 2008. The performance of the system is evaluated by using six real application codes. The experimental results indicate that the SX-9 system achieves a speedup of up to 7 compared to our previous NEC SX-7 system for the single-CPU sustained performance. In addition, the paper examines the effects of an on-chip vector cache named ADB on the performance, and confirms performance increases between 20 and 70% by selective caching on ADB.

1 Introduction

The Cyberscience Center (formerly Information Synergy Center) is one of seven national supercomputer centers in Japan. Since 1986, we have installed the latest vector supercomputer system at the time. Fig. 1 shows the improvements in performance and memory capacity of the five generations of supercomputer systems at Tohoku University. As shown, the latest system was installed every four to five years. Each achieves a performance improvement of 8 times or more in both Flop/s rate and memory capacity. In March 2008, we have installed a new supercomputer system named NEC SX-9 (see Fig. 2), which employs the first single 100+ GFlop/s

---

Hiroaki Kobayashi · Ryusuke Egawa · Hiroyuki Takizawa · Koki Okabe
Tohoku University, Sendai 980-8578, Japan,
e-mail: {koba, egawa, tacky, okabe}@isc.tohoku.ac.jp

Akihiko Musa
Tohoku University, Sendai 980-8578, Japan
NEC Corporation, Tokyo 108-8001, Japan, e-mail: musa@sc.isc.tohoku.ac.jp

Takashi Soga
NEC System Technologies, Osaka 540-8551, Japan, e-mail: soga-txa@necst.nec.co.jp

Yoichi Shimomura
NEC Software Tohoku, Sendai 980-0811, Japan, e-mail: y-shimomura@wx.jp.nec.com

vector processor. This paper gives an overview of our SX-9 system, the first machine of this kind delivered by NEC. We report its performance on real science and engineering applications, especially in terms of sustained performance of the 100+ Gflop/s vector processor and the effects of ADB, which is a newly designed on-chip cache for the SX vector processor.

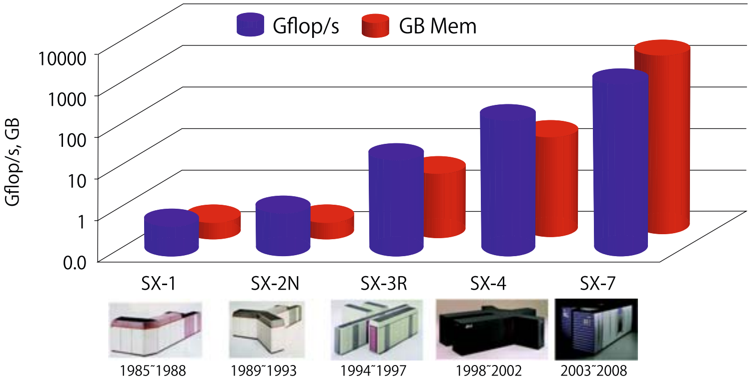


Fig. 1 Five Generations of NEC SX systems at Tohoku University

Fig. 2 NEC SX-9

2 System Overview

Fig. 3 shows the organization of our supercomputer systems. The SX-9 system consists of 16 nodes, each having 16 vector processors sharing a large memory space of 1 TB for shared memory processing (SMP). The 16 nodes are interconnected via a custom designed high-bandwidth crossbar network named IXS (Interconnect Crossbar Switch) at 256 Gbits (bi-directional). As each processor has a peak performance of 102.4 GFlop/s, the total performance of the system is 26.2 TFlop/s.

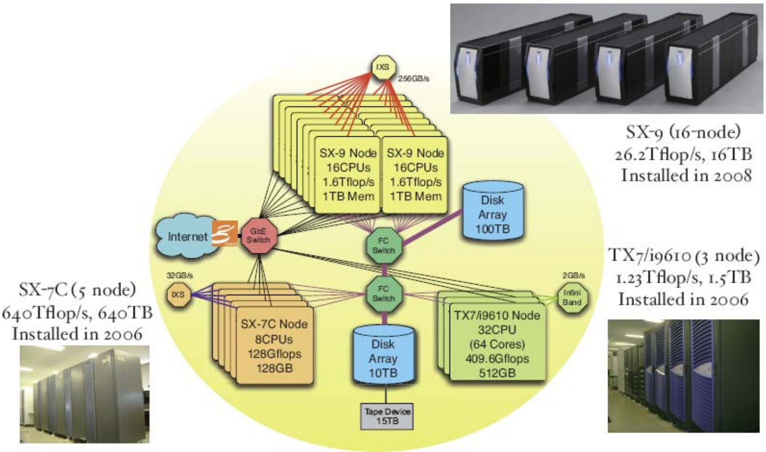


Fig. 3 Supercomputers at Tohoku University

Fig. 4 shows the architecture of the vector processor. The vector processor of SX-9 is designed in 65 nm technology and its operation frequency is 3.2 GHz. The SX-9 processor doubles both vector pipelines and vector units, and triples the clock frequency compared with the SX-7 vector processor (our previous system), resulting in a speedup of 11.6 for the single processor performance. The SX-9 processor also employs a newly designed 256 KB on-chip cache named ADB (Assigned Data Buffer) to support the vector load-store unit. By keeping vector data with locality in the ADB, a 4 Byte/Flop rate is guaranteed to effectively drive most of the vector pipes on the chip. Table 1 compares the performance of the SX-9 and the SX-7 system. The SX-9 offers a balanced combination of high performance vector CPUs and a larger shared main memory, which are connected with a higher memory bandwidth compared to the SX-7 system. This provides a user-friendly high-performance computing environment, in which users enjoy the potential of the system with less efforts.

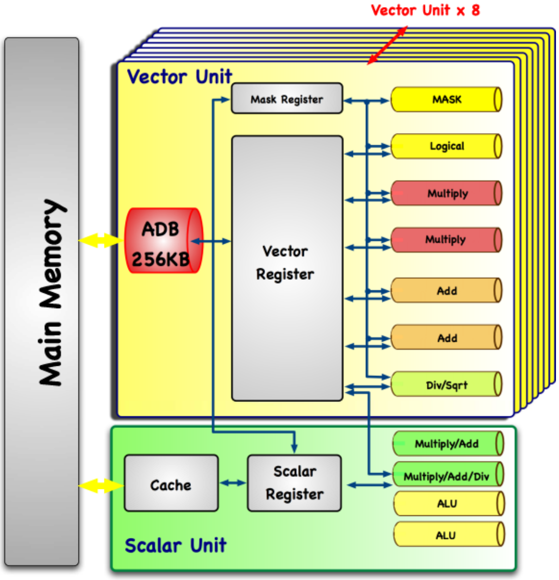


Fig. 4 SX-9 Processor Architecture

Table 1 Performance Comparison between SX-9 and SX-7

		SX-7 in 2003	SX-9 in 2008	Factor of Increase
per CPU	Freq.	1.1 GHz	3.2 GHz	2.9
	Vec. Perf.	8.83 GFlop/s	102.4 GFlop/s	11.6
	Mem. BW	35.3 GB/s	256 GB/s	7.3
per SMP node	Vec. Perf.	282 GFlop/s	1.6 TFlop/s	5.8
	Mem. Cap.	256 GB	1 TB	4.0
	Mem. BW	1.13 TB/s	4 TB	3.5
	Mem. Banks	16000	32000	2.0
	IXS BW	32 GB/s[a]	256 GB/s	8.0
per System	Total Perf.	2.1 TFlop/s	26.2 TFlop/s	12.5
	Total Mem.	2 TB	16 TB	8.0

[a] SX-7C(8)

3 Performance Evaluation

In this section, we examine the benchmark results of the SX-9 system. Table 2 shows six real applications that we have used for the performance evaluation. These applications are designed and developed by our users in the fields of science and engineering. In the performance evaluation, we are especially interested in the sustained performance of the 102.4 GFlop/s vector processor and the effects of the ADB on supporting vector load store operations.

Fig. 5 shows the sustained single CPU performance of the six benchmark programs for five NEC supercomputer systems, which are SX-9 (102.4 GFlop/s peak performance), SX-7 (8.8 GFlop/s peak performance), SX-8 (16 GFlop/s peak performance), SX-8R (35.2 GFlop/s peak performance) and TX7/i9610 ItaniumII (6.4 GFlop/s peak performance). The benchmark programs are compiled by the NEC Fortran compiler with automatic vectorization and automatic parallelization for the SX and TX7 systems. The SX-9 performs significantly better compared to the other systems, especially for the Antenna code. As the Antenna code has a high number of arithmetic operations per vector load/store, the enhanced pipelines and vector units as well as the higher clock frequency of the SX-9 linearly improve the sustained performance. The Plasma simulation requires a lot of list vector accesses and the gather/scatter operations put a large pressure on the load/store unit. The resulting sustained performance of the SX systems is not so high due to the bottleneck in memory accesses, but it is still impressive compared to ItaniumII.

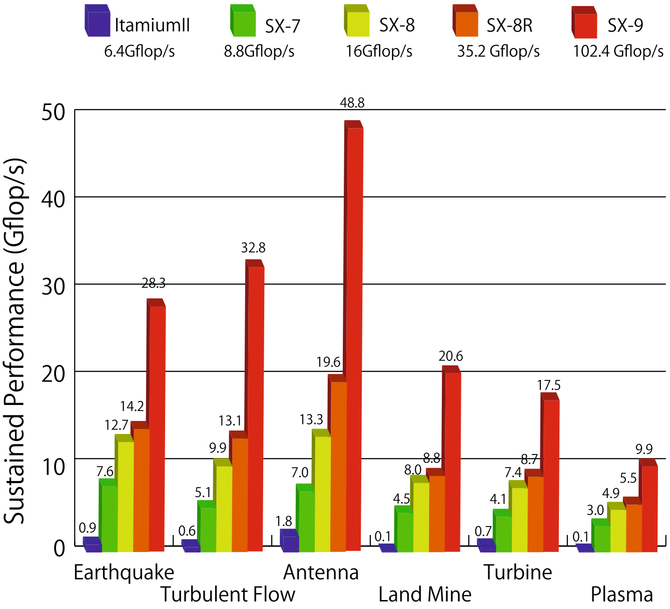
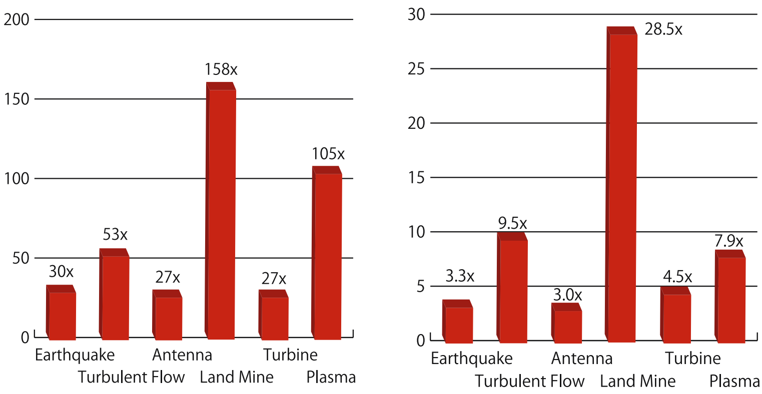


Fig. 5 Single CPU performance of six benchmarks

Fig. 6 depicts the direct comparison in the sustained performance between the SX-9 single processor and ItaniumII. As Fig. 6a shows, the SX-9 has a maximum speedup of 158 compared with a single ItaniumII CPU performance. Its average performance across all benchmarks is a factor of 67 higher as the ItaniumII even though the peak performance ratio between the two systems is only 16. Even in the case of a system with 16 ItaniumII CPUs, whose peak performance is the same as

Table 2 Benchmark Programs

Earthquake [1]	Simulation of seismic slow slip model.
Turbulent flow [7]	Direct numerical simulation of turbulent channel flow.
Antenna [5]	FDTD simulation of lens antenna using Fourier transform.
Land Mine [3]	FDTD simulation of array antenna ground penetrating radar for land mine detection.
Turbine [4]	Direct numerical simulation of unsteady flow through turbine channels for hydroelectric generators.
Plasma [2]	Simulation of upper hybrid wave in plasma using Lax-Wendroff method.



a vs. one ItaniumII CPU **b** vs. 16 ItaniumII CPUs

Fig. 6 SX-9 singe CPU performance vs. ItaniumII

for one CPU SX-9, the vector CPU achieves speedups of 3.0 up to 28.5 as shown in Fig. 6b. This significant difference in performance between SX-9 and ItaniumII is due to their memory bandwidths. For the Land Mine application, more than 95% of total execution time is spent for memory operations on ItaniumII because of its low cache hit rate of 70%. On the other hand, on NEC SX-9, only 25% of the total processing time results from memory access and the remaining 75% is available for arithmetic operations. The higher memory bandwidth of the SX-9 decreases the portion of memory processing in the total time and contributes to the higher sustained performance compared to the scalar system.

In the following, we examine the effects of the 256 KB ADB on the execution times of the applications. Fig. 7a shows one of the most time-consuming kernels of the Land Mine application. The Land Mine application is based on the FDTD method and has many data references with high locality. In this kernel, we selectively cache array `H_x` with the directive `!cdir ON_ADB(H_x)` as shown in Fig. 7b. By selective caching, vector loads of `H_x(i,j-1,k)` bring data into cache. This data will be reused subsequently for vector loads of `H_x(i,j,k)` and causes high cache hit rates. Fig. 8a shows the performance of SX-9 with ADB normalized by that without ADB. A performance improvement of 20% is obtained by ADB with selective caching. The Earthquake application also profits from the usage of the ADB. Fig. 9 displays one of the computationally most expensive kernels. If we selectively cache the loop-independent array `wary(j)` by pre-fetching it before loop, a performance improvement of 70% can be obtained as shown in Fig. 8b.

```
   DO 10 k=0,Nz
   DO 10 i=0,Nx
   DO 10 j=0,Ny
     E_x(i,j,k) = C_x_a(i,j,k)*E_x(i,j,k)
 &     + C_x_b(i,j,k) * ( (H_z(i,j,k)-H_z(i,j-1,k))/dy
 &     -(H_y(i,j,k)-H_y(i,j,k-1))/dz -E_x_Current(i,j,k) )
     E_y(i,j,k) = C_y_a(i,j,k)*E_y(i,j,k)
 &     + C_y_b(i,j,k) * ( (H_x(i,j,k)-H_x(i,j,k-1))/dz
 &     -(H_z(i,j,k)-H_z(i-1,j,k))/dx -E_y_Current(i,j,k) )
     E_z(i,j,k) = C_z_a(i,j,k)*E_z(i,j,k)
 &     + C_z_b(i,j,k) * ( (H_y(i,j,k)-H_y(i-1,j,k))/dx
 &     -(H_x(i,j,k)-H_x(i,j-1,k))/dy -E_z_Current(i,j,k) )
10 CONTINUE
```

a Original code

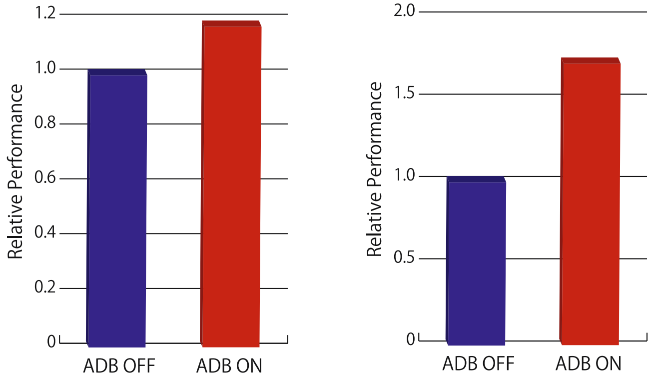
```
   DO 10 k=0,Nz
   DO 10 i=0,Nx
!cdir ON_ADB(H_x)
   DO 10 j=0,Ny
     E_x(i,j,k) = C_x_a(i,j,k)*E_x(i,j,k)
 &     + C_x_b(i,j,k) * ( (H_z(i,j,k)-H_z(i,j-1,k))/dy
 &     -(H_y(i,j,k)-H_y(i,j,k-1))/dz -E_x_Current(i,j,k) )
     E_y(i,j,k) = C_y_a(i,j,k)*E_y(i,j,k)
 &     + C_y_b(i,j,k) * ( (H_x(i,j,k)-H_x(i,j,k-1))/dz
 &     -(H_z(i,j,k)-H_z(i-1,j,k))/dx -E_y_Current(i,j,k) )
     E_z(i,j,k) = C_z_a(i,j,k)*E_z(i,j,k)
 &     + C_z_b(i,j,k) * ( (H_y(i,j,k)-H_y(i-1,j,k))/dx
 &     -(H_x(i,j,k)-H_x(i,j-1,k))/dy -E_z_Current(i,j,k) )
10 CONTINUE
```

b Code with the directive for selective caching

Fig. 7 Kernel of the Land Mine benchmark

4 Conclusions

In this paper we gave an overview of the new NEC SX-9 supercomputer installed at Tohoku University and examined its performance when executing science and engineering applications. Our system is the first one equipped delivered by NEC with 100+ GFlop/s vector chips. Various new technologies such as ADB (Assigned Data Buffer) are incorporated into it. The experimental results show remarkable



a Land Mine **b** Earthquake

Fig. 8 Effects of ADB

```
do i=1,ncells
do j=1,ncells
    wf_dip(i)=wf_dip(i)+gd_dip(j,i)*wary(j)
end do
end do
```

Fig. 9 Kernel of Earthquake

improvements for the considered applications. In this paper, only the single CPU performance of SX-9 has been discussed, but the scalable performance of our multi-node SX-9 system on a CFD MPI code has already been reported [6]. In our opinion, NEC SX-9 has the ability to become a key supercomputing platform.

Acknowledgements This work was part of a collaboration between Tohoku University and NEC, and many colleagues contributed to this project. We would also like to thank Professors Akira Hasegawa, Kunio Sawaya, Motoyuki Sato, Satoru Yamamoto, Yasuhiro Sasao, Associate professor Masahide Iizima of Tohoku University, Dr. Takahiro Tsukahara of Tokyo University of Science and Dr. Keisuke Ariyoshi of JAMSTEC for providing the benchmark codes.

References

1. Ariyoshi, K., Matsuzawa, T., Hasegawa, A.: The key frictional parameters controlling spatial variation in the speed of postseismic slop propagation on a subduction plate boundary. Earth and Planetary Science Letters **256**, 136–146 (2007)
2. Katoh, Y., Iizima, M.: A computer simulation study on the mode conversion process from slow X-mode to fast X-mode by the tunneling effect. In: Earth, Planets and Space, pp. e53–e56 (2006)
3. Kobayashi, T., et al.: FDTD simulation on array antenna SAR-GPR for land mine detection. In: Proceedings of SSR2003, pp. 279–283 (2003)