A case for PARAM Shavak: Ready-to-use and affordable supercomputing solution

Sandeep Agrawal, Shweta Das, Manjunatha Valmiki, Sanjay Wandhekar and Prof. Rajat Moona
High Performance Computing Technologies Group
Centre for Development of Advanced Computing (C-DAC)
Pune, India
sandeepa@cdac.in, shwetad@cdac.in, manjunathav@cdac.in, sanjayw@cdac.in, moona@cdac.in

***Abstract*— High Performance Computing (HPC) Systems are usually large systems which require specialized infrastructure. For a variety of small time users, who need performance of the parallel computing for their applications, such systems are unaffordable and inaccessible for a number of reasons. Even to setup a small state-of-the-art HPC system, such users would require vast efforts and expertise to design system specifications and to identify and install system software, tools and user applications. Also, going through such process would consume time and can be expensive.**

Clearly, there is a requirement of a small and low-cost ready-to-use HPC system which can be straightway put to utilization by end-users. In this paper, we present a case of a small, affordable and personalized supercomputing solution named PARAM Shavak [8, 9] which offers ready-to-use supercomputing-in-a-box solution based on commercial off-the-shelf HPC hardware resources. This solution is aimed as a support tool for research, design and development – often related to the education or small time designers. The solution is so architected that it provides scalability and power efficiency. We also discuss the uniqueness of our solution compared to several related initiatives which have been around and show its efficacy.

Keywords*— *PARAM shavak, affordable supercomputing, supercomputing in a box, low-cost supercomputer.

I. INTRODUCTION

There is a growing recognition of need of information technology, especially of High Performance Computing (HPC), for computational simulation and modeling which leads to better understanding and breakthroughs in areas of scientific and engineering applications [1]. HPC systems are usually large systems which incur huge operational and maintenance costs in addition to high capital cost [10]. Therefore, large HPC systems are deployed with specific objectives and deliverables. Access to such systems is granted to only limited set of users, especially those users whose research and development work is ready for adoption to large HPC facilities. Similarly, the access to medium scale HPC systems is again limited to within a department or to groups within an institution, and the systems are not accessible to a variety of small time users. However, even when the systems might be accessible to small time users, these are available only in a shared mode. Under such small workloads being shared with large workloads, often the average waiting time for the job becomes substantial even when the execution time may be small. Large waiting time is not desirable for short jobs and exploring users. Also, as large HPC systems are generally meant for production runs, they are not suitable for HPC learners, amateur developers and casual users. Small systems are conducive to get familiarized with HPC environment, develop and test applications, run small production jobs and to get quick turnaround.

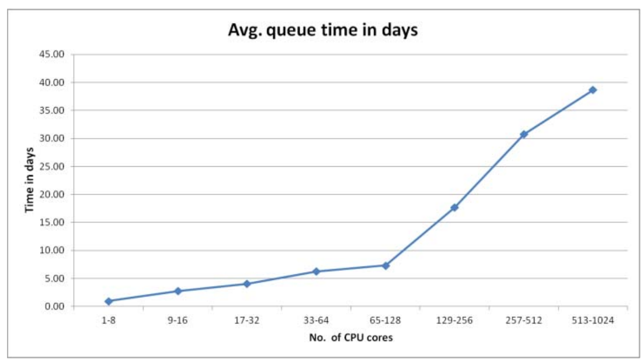

Figure 1. Avg. queue time vs number of CPU cores for the jobs on PARAM Yuva II

Specifications of PARAM Yuva II: Rank 447 [23] with peak and sustained compute power of 520.4TF and 388.4TF; 2x Xeon E5-2670 8Cores 2.600GHz, Infiniband FDR, Intel Xeon Phi 5110P

Several large facilities exist in India [24] but these facilities remain out of reach for most of the small users. Centre for Development of Advanced Computing (C-DAC) [12] maintains several large HPC systems such as PARAM Yuva II [11]. The average queue time is plotted with reference to the number of CPU cores used by various jobs on PARAM Yuva II for six month duration in Fig.1. It can be seen in the figure that the average waiting time increases with the number of processing cores requested by the job. Similar patterns of waiting time are reported by other large HPC installations [21]. In such installations, a job requiring small number of cores (say 16) for only a few minutes of execution time would have to typically

978-1-5386-3250-5/17 $31.00 © 2017 IEEE
DOI 10.1109/HPCS.2017.66

wait for a number of days in the queue before getting scheduled for execution. Therefore, users with small jobs face unusual delays and get intermittent focus on their applications. Restricted accessibility of HPC systems keep several such potential HPC users distant from the systems leading to smaller HPC communities. Such users would benefit from small systems which are accessible on demand. The need for accessible supercomputers has always been felt by the HPC community [27].

Researchers or developers seeking to customize or design and develop their parallel applications often need interactive access to the HPC system to debug applications, collect statistics by monitoring utilization of resource to fine tune implementations. However, interactive access to large HPC systems is allowed through batch processing systems which may provide requested resources at a later time, making it difficult for software design and development. Hence, small systems with flexibility of interactive access and equipped with similar hardware, software development and runtime environment as of large systems would be more relevant for the use case. Once such applications are proven, they can be scaled on bigger HPC systems.

Large HPC systems have their own fallouts in terms of need for skilled manpower for system administration as well as to keep pace with the fast technological advancements. HPC skilled system administrators are scarce resource and large HPC systems need a dedicated team of such administrators. As huge costs are incurred to build and maintain large HPC systems their life span is much more with respect to the cycle of technological advancements leading to use of older generation of computational resources for a long time. Even, up-gradation of such systems with new accelerator devices, operating system and other software is fairly difficult owing to the binary compatibility of existing codes of the huge user base. While a small and low-cost system can be relatively easily upgraded or even replaced, especially making more sense for educational and exploration kinds of need, to keep its users abreast of the latest tools and technologies.

It is therefore felt that there is a substantial requirement of smaller and affordable HPC systems which are readily available on demand and which provide similar programming and runtime environment as of the large HPC systems. In this paper, we present a small, affordable and personalized supercomputing solution, named PARAM Shavak, which offers low-cost supercomputing-in-a-box solution. It is a ready to use HPC system with development and runtime environment of a state-of-the-art supercomputing system. The solution is designed so as to achieve scalability and power efficiency.

The rest of this paper is organized as follows. In section II, we present various existing solutions for affordable and accessible supercomputing. In section III, we present the details of PARAM Shavak architecture and bring forth the uniqueness of this solution while comparing with other similar initiatives world-wide. We also present the performance of this machine with various benchmarks of HPC applications. We conclude this work in section IV, followed by future work briefed in section V.

II. RELATED WORKS

There have been several initiatives to proliferate use of HPC among small users or users with specialized requirements. These initiatives range from building a virtual cluster to development of specialized supercomputers.

The need for affordable and accessible supercomputing has been in demand for several decades. Greenberg et al. in their paper [25] outlined the need for department-scale supercomputing. Recently, with an objective of attracting newer scientific areas for HPC and bringing corresponding scientists and researchers who are otherwise scantly involved in HPC computations, the HPC-Europa2 project [13] built a live DVD through which a regular computer can be bootstrapped into a virtual cluster under a hypervisor. This environment then provides a platform to learn parallel programming with state-of-the-art tools and libraries. As a result, the HPC-Europa2 live DVD provides an e-learning and teaching platform for HPC.

Nvidia DGX-1 [14] is a supercomputer-in-a-box system specifically built for deep learning by nVIDIA Corp. Math Supercomputer-In-A-Box [14] developed by Advanced Cluster System LLC uses Mac Pro based cluster approach to provide supercomputing solution using Mathematica [26] software from Wolfram Research [26].

Affordable high-performance cluster systems had been designed and implemented based on the PlayStation 3 (PS3) Cell microprocessors [39] with total cost being very less compared to standard HPC cluster. A few independent initiatives with the networked PS3s concluded their usefulness in using them as HPC platform for computer science courses, as well as to perform simulations such as protein folding, astrophysical simulations. However due to security concerns, in 2010 Sony announced it would be disabling the ability to run other operating system with the v3.21 update. Similarly, using a lightweight Linux operating system called Raspbian a cluster has been implemented with Raspberry Pi for example Tiny Titan system at ORNL [40]. Some application examples on this cluster include a 2D smoothed hydrodynamic particle calculation module and a parallel computing code to calculate a Mandelbrot fractal.

Such initiatives have either created HPC solutions for specific application areas, rather than for general purpose computing, or have facilitated emulation of HPC environment. As we analyze the fallouts of these initiatives, we find a void for ready-to-use and accessible HPC solution based on state-of-the-art HPC environment including hardware and software, for general purpose computing while being affordable. With its integrated approach to provide pre-installed HPC software development and execution environment on industry standard HPC hardware, PARAM Shavak precisely fills this void, and is well suited for learners and for expert users as a personal supercomputer.

III. OVERVIEW OF PARAM SHAVAK

In order to enable wider accessibility of HPC platform to potential researchers and learners, we provide an affordable ‘Supercomputing-in-a-box’ solution, christened as PARAM Shavak.

PARAM Shavak is a table-top based supercomputing-in-a-box solution which does not require any data centre ecosystem. The compact nature of its design enables perfect bonding between HPC requirements and infrastructural availability, making it an ideal solution both for personal supercomputing as well as for training. This supercomputing-in-a-box solution intends to provide computational resource with advanced technologies to perform high-end computations on a moderate scale for the scientific, engineering and academic programs. PARAM Shavak is a ready-to-use system with HPC software development and runtime environment preinstalled.

The PARAM Shavak system is powered with a minimum of 2 multi-core x86_64 CPUs, each with 12 or more cores, currently being Intel Xeon Haswell CPUs, along with up to two industry-standard accelerator cards such as Intel Xeon Phi [15] or Nvidia Tesla GPGPU [16]. PARAM Shavak offers power efficient system by providing 3.8TF peak, with 2x K40 [16] GPGPU devices, in a power envelope of 820W and, also by providing 3.4TF peak for 950W with 2x Intel Xeon Phi 7120A [36] devices. Other important specifications include min. 64GB RAM DDR4 2133MHz, CentOS, 8 TB RAID 5 storage, 2x 1GbE n/w ports. The simplest form of PARAM Shavak, without any accelerator, provides 804 GFLOPS of sustained performance on LINPACK [28] benchmark. The performance scales well with accelerators.

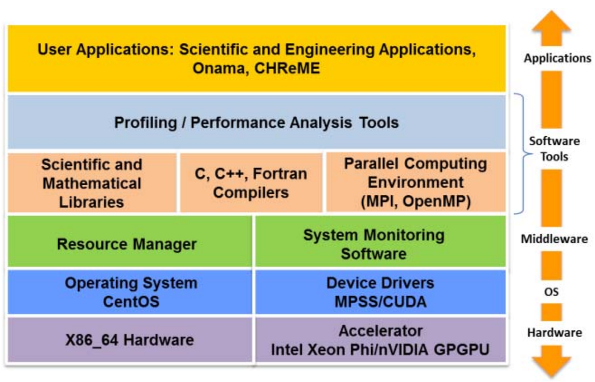

Figure 2. PARAM Shavak Architecture Block Diagram

PARAM Shavak (Fig. 2) is pre-loaded with parallel software development tools which include parallel programming environment, parallel scientific and mathematical libraries, application specific libraries, language compilers, code profilers, analyzers, debuggers, vectorization advisor, accelerator libraries and parallel applications that use accelerators such as GPGPU and Xeon Phi. The system also has the resource manager software for job submission and management as available on large HPC systems. This setup allows users to use the system in interactive mode as well as in batch mode. PARAM Shavak can be accessed remotely by users present in a common network.

The PARAM Shavak solution is also equipped with C-DAC's indigenously developed C-DAC's HPC Resource Management Engine (CHReME) [8, 9, 17] and Onama [8, 9, 18] software. CHReME is a web based HPC portal which brings a level of interactivity for users and relieves them from the command line mode of interaction with the system. The portal provides several facilities, through which users can create job scripts, submit and manage jobs, use scientific applications specific interfaces, configure execution environment, monitor resources utilization and perform basic system administration tasks. CHReME is a recipient of IDC HPC Innovation Excellence Award [37] announced at the International Data Corporation (IDC) briefing on the sidelines of Supercomputing Conference 2011 held at Seattle. Onama is a software tool which consists of several well selected set of parallel and serial applications and tools across various engineering areas like mechanical, civil, computer, chemical etc. Onama also comprises of numerical packages, commonly used tools and tutorial parallel programs, with source codes, based on MPI, OpenMP, numerical algorithms and mathematical libraries in C and FORTRAN languages. Onama is a GUI based execution model which allows execution of integrated applications with minimal effort and consists of sample data sets and applications' manuals. In Fig. 3, we present submission of a CFD application batch job through Onama. Onama is a recipient of Manthan Award 2013 in the categoryof e-Science and Technology [38]. Both Onama and CHReME are integrated with resource manager so that all the jobs are submitted through the scheduling software. Further details can be found in papers published on Onama [34] and CHReME [35].

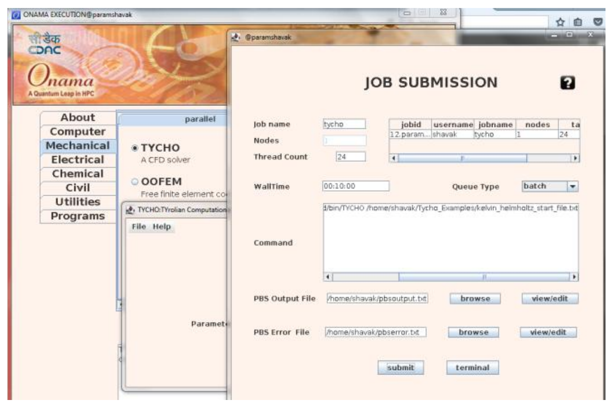

Figure 3. Onama

PARAM Shavak makes it easier for its scientific users to focus on their research and development work rather than to focus on complex procedures of installation of scientific HPC applications, especially with use of accelerators. PARAM Shavak comprises of pre-compiled open-source HPC applications across molecular dynamics, computational fluid dynamics, bioinformatics, atmospheric science and materials science domains. These pre-loaded HPC applications are compiled in a manner so as to optimally utilize the advanced capabilities, such as vectorization, of the underlying processor and accelerator device. The solution also offers bash script based interactive execution of HPC applications to enable learners develop fast understanding of the various steps involved in the execution of the parallel applications. Also, the application source codes are provided which in our opinion is an approach to enable education for HPC.

```
*** OpenFOAM v4.0 application ***

Please select an option:
 1. Application execution with sample dataset
 2. Environment setup
 3. Commonly used commands
 4. Performance benchmarks
 5. Visualize output using ParaView
 6. How to run through resource manager
 0. EXIT: press 0

Enter choice: 3
===============
***Commonly used commands for OpenFOAM v4.0 application***

Command for mesh generation, for example, for case smallPoolFire3D:
$ blockMesh -case smallPoolFire3D

Command to split mesh into many regions:
$ topoSet -case smallPoolFire3D

Command to create a patch of input string:
$ createPatch -case smallPoolFire3D -overwrite

Command to decompose mesh for parallel run:
$ decomposePar -case smallPoolFire3D -force

Command to run fireFoam on the mesh in parallel:
$ mpiexec.hydra -n 24  fireFoam -case smallPoolFire3D  -parallel
```

Figure 4. Snippet of execution of tutorial bash script for OpenFOAM

In Fig. 4, we present a snippet of tutorial script for interactive execution of OpenFOAM [31] application, a CFD software to solve complex fluid flows, acoustics, solid mechanics, electromagnetics etc. related problems. The script consists of several sections such as application execution with sample dataset, environment setup, commonly used commands, performance benchmarks, output visualization, running through resource manager etc. The application execution section invokes a sample run of OpenFOAM using its smallPoolFire3D example, demonstrating various important steps such as mesh generation, splitting of mesh into regions, mesh decomposition for parallel run, parallel execution for the case. Finally, the output can be analysed and visualized through ParaView, which again is installed with parallel visualization capabilities.

In the area of atmospheric science PARAM Shavak provides the Weather Research and Forecasting (WRF [32]) model which is widely used for atmospheric research and operational forecasting. For a beginner it is a lengthy cycle to use this model from data pre-processing to post-processing. Our command-line based tutorial script helps users to understand how to invoke commands and set important parameters for different stages of model execution. For example, while running this interactive script with the sample dataset provided in PARAM Shavak, users can understand the usage of model starting from defining their geographical domain, by inputting latitude and longitude, to the parallel execution of main model and visualization of data. In Fig. 5 we show an output visualization using GrADS [33] tool, invoked through the script.

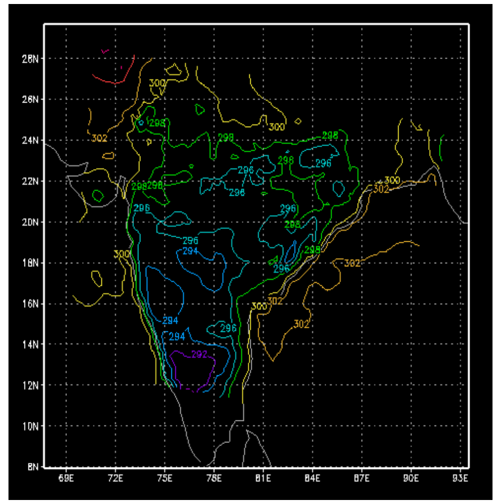

Figure 5. Data visualization through tutorial bash script for WRF

In the area of atmospheric science PARAM Shavak provides the Weather Research and Forecasting (WRF [32]) model which is widely used for atmospheric research and operational forecasting. For a beginner it is a lengthy cycle to use this model from data pre-processing to post-processing. Our command-line based tutorial script helps users to understand how to invoke commands and set important parameters for different stages of model execution. For example, while running this interactive script with the sample dataset provided in PARAM Shavak, users can understand the usage of model starting from defining their geographical domain, by inputting latitude and longitude, to the parallel execution of main model and visualization of data. In Fig. 5 we show an output visualization using GrADS [33] tool, invoked through the script.

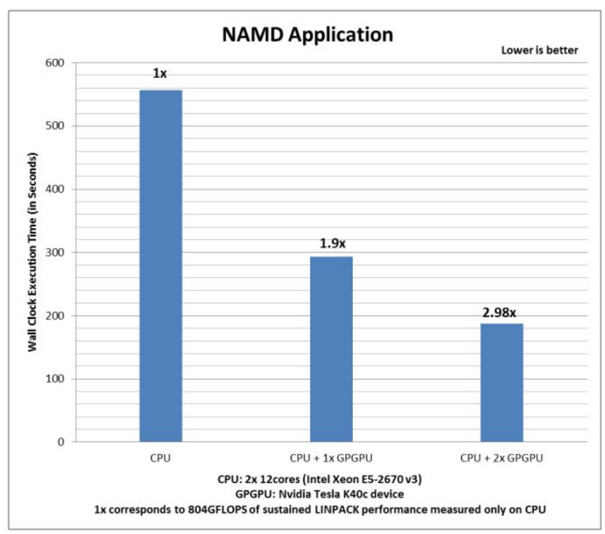

Figure 6. NAMD scalability on GPGPU based PARAM Shavak

In Fig. 6, we show scalability of NAMD [20] application, a molecular dynamics simulation code, on GPGPU based PARAM Shavak. With respect to performance on CPU-only, total 24 cores, as the baseline performance, the application scales up to 1.9x and 2.98x by utilizing one and two Tesla K40c GPGPU accelerator devices respectively along with the use of all CPU cores. The benchmark substantiates that with addition of the accelerator devices the overall performance of the simulation is approximately tripled, whereas both the capital cost and operating cost are only modestly increased, leading to a much better computational performance to cost ratio. The dataset used is 1,066,628 atoms based STMV (virus) benchmark which is provided as an example simulation on the application software website. It was run for 1000 timesteps. The performance benchmark is performed through the tutorial bash script for NAMD. The script provides the choice to run the application either on one or two accelerator devices, GPGPU or Xeon Phi, or only on CPU cores along with a host of other options.

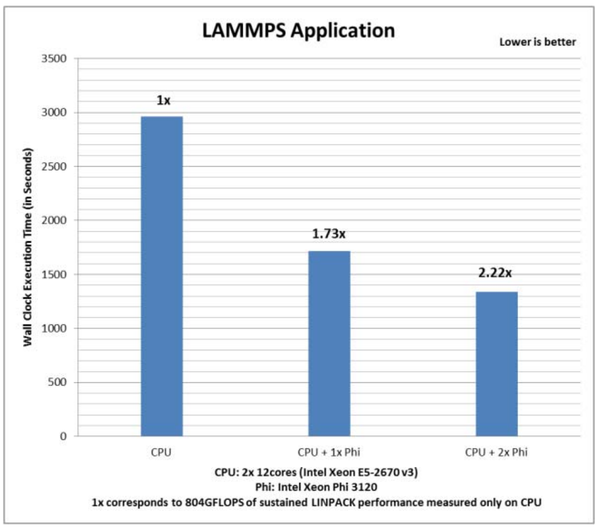

Figure 7. LAMMPS scalability on Xeon Phi based PARAM Shavak

In Fig.7, we present scalability of another molecular dynamics simulation code, LAMMPS [19], on Xeon Phi accelerator based PARAM Shavak. The application scales up to 1.73x by utilizing the host CPUs and one Xeon Phi 7120A accelerator device. Performance gain continues to hold up to 2.22x with two Xeon Phi 7120A accelerator devices and the CPUs compared to CPUs-only baseline performance. The dataset used is LJ, atomic fluid, Lennard-Jones potential with 32,000 atoms, as available in the bench directory of the LAMMPS distribution. It was run for 100,000 timesteps.

PARAM Shavak comprises of extensive reference material, user manuals, video and text tutorials based on parallel architectures, parallel programming models, parallelization approaches, applications etc to educate users. Till date we have deployed over 30 PARAM Shavak all over India and they have been well accepted by users' base of engineering students, faculties, researchers and scientists. Users of PARAM Shavak have recognized it as an affordable and ready-to-use supercomputing-in-a-box solution. Many faculties have been using PARAM Shavak to educate their students about parallel programming and to apply such techniques to write parallel codes for the problems given under workshop tutorials. Researchers and scientists have been using PARAM Shavak to conduct their numerical simulations using HPC applications pre-installed on the system. Also, these users can install HPC applications not available on the system and can seek help from the support team of PARAM Shavak. Users new to HPC have find it simple to use thereby avoiding large efforts to learn the usage of HPC and without requiring extensive expertise to manage an HPC system.

IV. CONCLUSION

In this paper, we have tried to establish the requirement of small, affordable and pre-installed state-of-the-art HPC systems which can be readily put to utilization among users with moderate HPC needs. Though, several initiatives worldwide have developed solutions to cater to demands of such users, but these solutions have lacked in one or more of the following aspects which includes state-of-the-art computing environment, plug-and-play, general purpose usability, affordability, learning curve and no requirement of datacenter. PARAM Shavak is an engineering solution to address perennial academic problem of lack of HPC capability and capacity. PARAM Shavak is offered as a pre-installed HPC server integrated with indigenous tools, video and text tutorials to reduce the learning curve of new users, while advanced users can readily start using pre-compiled HPC applications.

Experiences of users of PARAM Shavak have indicated its usefulness in research areas such as molecular dynamics, CFD, physics, biotechnology etc. Further many of them find it a great tool for HPC education to their students.

V. FUTURE WORK

It will be our constant endeavour to keep PARAM Shavak at par with the latest trends, technologies and user applications in the area of HPC. We are also in the process of building an update server through which patches and upgrades of applications, libraries, compilers etc. can be effected on the deployed PARAM Shavak systems. Based on inputs from users of PARAM Shavak, we see many users are interested in exploration of big data applications. Also, as big data and HPC are witnessing increasing commonality in terms of infrastructure and their applications, we are working on to build PARAM Shavak solution for big data platform.

ACKNOWLEDGMENT

Authors would like to acknowledge the HPC-Technologies Group at C-DAC and in particular the individuals who have contributed to the development of PARAM Shavak, viz., Gulshan Khatri, Pravin Pawar, Puneet Singh, B. S. V Ramesh, Ashish Ranjan, Mohammed Sajeed, Nirmal Kumar, Sucheta Pawar, Vishal Bahedia, Tarun Singh, Chitranjan Singh, Nisha Kurkure, Roshan Tayde, Prashant Dinde, Deepu C.V, Deepak Gaud, Abhishek Das, Prasad Wadlakondawar and Sudhir Chavan. Authors would also like to thank and place a deep gratitude to National PARAM Supercomputing Facility team at C-DAC for helping with information on running of a large HPC installation. Authors would also like to acknowledge various PARAM Shavak user groups at PSG College of Technology Coimbatore [29], Kerala Technological University Thiruvananthapuram and Manipal University Jaipur [30] for

providing their insights into the usage, accessibility and affordability of PARAM Shavak.

REFERENCES

[1] Franck Cappello, Wolfgang Gentzsch, Mateo Valero, "The Era of Exascale Sciences: Challenges, Needs and Requirements" in The 2013 International Conference on High Performance Computing and Simulation, Helsinki, July 2013.

[5] OpenMP Architecture Review Board, "The OpenMP API specification for parallel programming", http://openmp.org/wp/about-openmp/, as on June 2016.

[6] Eduard Ayguade, Nawal Copty, Alejandro Duran, Jay Hoeflinger, Yuan Lin, Federico Massaioli, Xavier Teruel, Priya Unnikrishnan, Guansong Zhang, "The Design of OpenMP Tasks," IEEE Transactions on Parallel and Distributed Systems, vol. 20, pp. 404 - 418, June 2008.

[7] NVIDIA Corporation, "CUDA Parallel Computing Platform", http://www.nvidia.com/object/cuda_home_new.html, as on June 2016.

[8] C-DAC, "World's first compact and energy efficient supercomputer launched by C-DAC", http://cdac.in/index.aspx?id=pk_pr_prs_rl223, as on June 2016.

[9] C-DAC, "PARAM Shavak - Supercomputing Solution in a Box", http://www.cdac.in/index.aspx?id=hpc_ss_param_shavak, äs on June 2016.

[10] Dale Sartor, Mark Wilson, "Money for Research, Not Energy Bills: Finding Energy and Cost Savings in High-Performance Computer Facility Designs," IEEE Computing in Science & Engineering, vol.12, pp. 11-19, Nov-Dec. 2010.

[11] C-DAC, "PARAM Yuva II", http://www.cdac.in/index.aspx?id=hpc_nsf_param_yuva_ii, äs on June 2016

[12] Centre for Development of Advanced Computing (C-DAC), http://www.cdac.in/, as on June 2016.

[13] Anthony Sulistio, Alexander Schulz, Rainer Keller, Panagiotis Kritikakos, Mourtzoukos Kostas, Theodora Varvarigou, "Building a virtual cluster using HPC-Europa2 live DVD" in Proceedings of International Conference on High Performance Computing and Simulation (HPCS), Madrid, July 2012.

[14] NVIDIA Corporation, "NVIDIA DGX-1: Deep Learning System", http://www.nvidia.com/object/deep-learning-system.html, as on June 2016.

[15] Intel Corporation, "Intel Xeon Phi Coprocessor", https://software.intel.com/en-us/mic-developer, as on June 2016.

[16] NVIDIA Corporation, "NVIDIA TESLA GPU ACCELERATORS", http://www.nvidia.com/content/tesla/pdf/nvidia-tesla-kepler-family-datasheet.pdf, downloaded Feb 2017

[17] C-DAC, "CHReME - C-DAC HPC Resource Management Engine", https://cdac.in/index.aspx?id=hpc_hpcs_chreme, as on Feb 2017

[18] C-DAC, "ONAMA: A quantum leap in HPC", https://cdac.in/index.aspx?id=pdf_onama_brochure, as on Feb 2017

[19] LAMMPS for Intel Xeon Phi Coprocessor, https://software.intel.com/en-us/articles/lammps-for-intel-xeon-phi-coprocessor, as on June 2016.

[20] James C. Phillips, Rosemary Braun, Wei Wang, James Gumbart, Emad Tajkhorshid, Elizabeth Villa, Christophe Chipot, Robert D. Skeel, Laxmikant Kale, and Klaus Schulten. Scalable molecular dynamics with NAMD. Journal of Computational Chemistry, 26:1781-1802, 2005. NAMD was developed by the Theoretical Biophysics Group in the Beckman Institute for Advanced Science and Technology at the University of Illinois at Urbana-Champaign, http://www.ks.uiuc.edu/Research/namd/

[21] Daniel Nurmi, Anirban Mandal, John Brevik, Chuck Koelbel, Rich Wolski, Ken Kennedy, "Evaluation of a Workflow Scheduler Using Integrated Performance Modelling and Batch Queue Wait Time Prediction" in SC 2006 Conference, Proceedings of the ACM/IEEE, Nov. 2006

[2] Carlos Boneti, Roberto Gioiosa, Francisco J, Cazorla, Mateo Valero, "A dynamic scheduler for balancing HPC applications" in 2008 SC - International Conference for High Performance Computing, Networking, Storage and Analysis, November 2008.

[3] Blaise Barney, "Message Passing Interface (MPI)", https://computing.llnl.gov/tutorials/mpi/, as on June 2016.

[4] E. Lusk, "MPI in 2002: has it been ten years already?" in Proceedings of IEEE International Conference on Cluster Computing, Chicago, Sep 2002.

[22] Tuan V. Dinh, Lachlan L. H. Andrew, Philip Branch, "Exploiting per user information for supercomputing workload prediction requires care" in Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium, May 2013

[23] Jack Dongarra, Knoxville, Horst Simon, Martin Meuer, Eric Strohmaier, Hans Meuer,Top500 List, http://top500.org/, seen on 27 Feb, 2017

[24] Sathish Vadhiyar, Yogesh Simmhan, "Top Supercomputers in India", maintained by Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore, http://topsupercomputers-india.iisc.ernet.in/, seen on 15 June, 2016

[25] David S. Greenberg , William E. Hart , Cynthia A. Phillips,"Enabling Department-Scale Supercomputing", http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.30.8762&rep=rep1&type=pdf, downloaded on 15 June,2016

[26] Wolfram Research, WOLFRAM MATHEMATICA, https://www.wolfram.com/mathematica/, seen on 4 June 2016

[27] W. Leinberger, G. Karypis, V. Kumar, "Job Scheduling in the presence of Multiple Resource Requirements" Supercomputing, ACM/IEEE 1999 Conference, Nov. 1999

[28] Jack Dongarra, "FAQs on the Linpack Benchmark and Top500", http://www.netlib.org/utk/people/JackDongarra/faq-linpack.html, seen on Feb 2017

[29] Dr. A Chitra, PSG College of Technology, Coimbatore, E-mail, Mar 2016

[30] Mr. Rahul Saxena, School of Computing and Information Technology(SCIT), Manipal University, Jaipur, E-mail, 16 Jun 2016

[31] OpenCFD Ltd. (ESI Group), "About OpenFOAM", http://www.openfoam.com/, as on Feb 2017

[32] Skamarock, W. C., J. B. Klemp, J. Dudhia, D. O. Gill, D. M. Barker, M. G Duda, X.-Y. Huang, W. Wang, and J. G. Powers, 2008: A Description of the Advanced Research WRF Version 3. NCAR Tech. Note NCAR/TN-475+STR, 113 pp. doi:10.5065/D68S4MVH

[33] George Mason University, "Overview of GrADS", http://cola.gmu.edu/grads/, seen on Feb 2017

[34] G. Misra, S. Agrawal, N. Kurkure, S. Das, K. Mathur, S. Pawar, "ONAMA: A Quantum Leap in High Performance Computing", Advanced Materials Research, Vols. 328-330, pp. 2337-2342, 2011, DOI: 10.4028/www.scientific.net/AMR.328-330.2337

[35] HPC-UA'2011, "CHReME: A Web Based Application Execution Tool for using HPC Resources", http://hpc-ua.org/hpc-ua-11/files/proceedings/1.2(19).pdf, 2011

[36] Intel Corporation, "Intel® Xeon Phi™ Coprocessor 7120A", http://ark.intel.com/products/80555/Intel-Xeon-Phi-Coprocessor-7120A-16GB-1_238-GHz-61-core, seen on Feb 2017

[37] International Data Group (IDG), "IDC Announces New Winners of HPC Innovation Excellence Awards", https://www.idg.com/news/idc-announces-new-winners-of-hpc-innovation-excellence-awards/, seen on Feb 2017

[38] Manthan Award (Digital Empowerment Foundation), "e-Science & Technology, Winner 2013", http://manthanaward.org/e-science-technology-winner-2013, as seen on Feb 2017

[39] Wikipedia, "PlayStation 3 cluster", https://en.wikipedia.org/wiki/PlayStation_3_cluster, as seen on Feb 2017

[40] Hpcwire, "Tiny Titan Preps Students for Multicore Era", https://www.hpcwire.com/2014/06/16/tiny-titan-preps-students-multicore-era/, as seen on Feb 2017