ppOpen-HPC: Open Source Infrastructure for Development and 
Execution of Large-Scale Scientific Applications on 
Post-Peta-Scale Supercomputers with Automatic Tuning (AT)

Kengo Nakajima
Information Technology Center, The University of Tokyo
nakajima@cc.u-tokyo.ac.jp

Abstract
We propose an open source infrastructure for development and 
execution of optimized and reliable simulation codes on post-peta-
yscale (pp) parallel computers with heterogeneous computing 
nodes which consist of multicore CPU’s and accelerators., named 
"ppOpen-HPC". ppOpen-HPC consists of various types of librar-
ies, which covers various types of procedures for scientific com-
putations. Source code developed on a PC with a single processor 
is linked with these libraries, and generated parallel code is opti-
mized for post-peta-scale system. Capability of automatic tuning 
is important and critical technology for further development on 
new architectures and maintenance of the framework.

Categories and Subject Descriptors G.1.10 [Numerical Analy-
sis]: Applications

General Terms Algorithms, Performance

Keywords Scientific Computing; Parallel Numerical Library; 
Automatic Tuning

1. Introduction

Recently, high-end parallel computer systems are becoming 
larger and more complex. It is very difficult for scientists and 
engineers to develop efficient application codes, which make use 
of potential performance of these systems.

We propose an open source infrastructure for development and 
execution of optimized and reliable simulation codes on large-
scale parallel computers. This infrastructure is named “ppOpen-
HPC (http://ppopenhpc.cc.u-tokyo.ac.jp/)”, where “pp” stands for 
"post-peta-scale", as shown if Fig.1 [1].

Target system is post-peta-scale supercomputer system with 
heterogeneous computing nodes which consist of multicore 
CPU’s and co-processors/accelerators, such as GPU’s and many-
cores (e.g. Intel MIC). Peak performance is O(10) PFLOPS, and 
number of cores is O(10⁶). Target system will be installed at In-
fomation Technology Center, The University of Tokyo 
(ITC/U.Tokyo) [2] in FY.2014 or FY.2015 (Fig.2). ppOpen-HPC 
supports more than 1,500 users of supercomputer systems of 
ITC/U.Tokyo to switch from homogeneous multicore clusters to 
post-peta-scale system with heterogeneous computing nodes.

"ppOpen-HPC" is five-year project (FY.2011-2015) and a 
part of “Development of System Software Technologies for Post-
Peta-Scale High Performance Computing” funded by JST/CREST 
(Japan Science and Technology Agency, Core Research for Evo-
lutional Science and Technology).

The infrastructure consists of various types of libraries for sci-
centific computations. Source code developed on a PC with a sin-
gle processor is linked with these libraries, and generated parallel 
code is optimized for post-peta-scale system. The framework 
covers various types of procedures for scientific computations,
such as parallel I/O of data-sets, matrix-formation, linear-solvers with practical and scalable preconditioners, visualization, adaptive mesh refinement and dynamic load-balancing, in various types of computational models, such as FEM (Finite-Element Method), FDM (Finite-Difference Method), FVM (Finite-Volume Method), BEM (Boundary Element Method), and DEM (Discrete Element Method) (Fig.3).

This type of framework will provide dramatic efficiency, portability, and reliability in the development and execution of scientific applications. It reduces both of number of steps of the source code and the duration for parallelization and optimization of legacy codes.

Automatic tuning (AT) technology enables automatic generation of optimized libraries and applications under various types of environments.

In ppOpen-HPC, these five issues are carefully considered. ppOpen-HPC includes following four components, as shown in Fig.1:

- ppOpen-APPL
- ppOpen-MATH
- ppOpen-AT
- ppOpen-SYS

Libraries in ppOpen-APPL, ppOpen-MATH, and ppOpen-SYS are called from user’s programs written in Fortran and C/C++ with OpenMP. OpenACC will be also supported. All issues related to hybrid parallel programming, CUDA, OpenCL are “hidden” from users by ppOpen-AT.

2. Overview of ppOpen-HPC

“ppOpen-HPC” is carried out by following six institutes, which include four departments of the University of Tokyo:

- The University of Tokyo
- Information Technology Center (ITC)
- Atmospheric and Ocean Research Institute (AORI)
- Center for Integrated Disease Information Research (CIDIR)
- Research into Artifacts, Center for Engineering (RACE)
- Center for Integrated Disease Information Research (CIDIR)
- Japan Agency for Marine-Earth Science and Technology (JAMSTEC)

Expertise of members covers wide range of disciplines related to scientific computing, such as system software, numerical library/algorithm, computational mechanics and earth sciences.

Five key issues for algorithms and applications towards post-peta/exa-scale computing are as follows:

- Hybrid/Heterogeneous Architecture (Multicore + GPU, Multicore + Manycore)
- Mixed Precision Computation
- Auto-Tuning/Self-Adapting
- Fault Tolerance
- Communication Reducing Algorithms

2.1 ppOpen-APPL

ppOpen-APPL is a set of libraries corresponding to each method (FEM, FDM, FVM, BEM, DEM), which include I/O, optimized linear solvers, matrix assembling, AMR etc., as shown in Fig.1.

ppOpen-APPL is a set of libraries which covers various types of procedures for scientific computations, such as parallel I/O of data-sets, matrix-formation, linear-solvers with practical and scalable preconditioners, visualization, adaptive mesh refinement (AMR) and dynamic load-balancing, in various types of models, such as FEM, FDM, FVM, BEM, and DEM, as shown in Fig.3.

Each component should be developed based on existing practical application codes. ppOpen-APPL provides common data structures and interfaces based on parallel netCDF, which will support users to easily implement procedures of ppOpen-HPC on legacy codes.

2.2 ppOpen-MATH

ppOpen-MATH consists of common numerical libraries, such as multigrid solvers (ppOpen-MATH/MG) (Fig.4), parallel graph libraries (ppOpen-MATH/GRAPH), parallel visualization (ppOpen-MATH/VIS), and library for coupled multi-physics simulations (ppOpen-MATH/MC) (Fig.5).

If we consider the improvement of multigrid process for post-peta/exa-scale systems, it is important to develop new methods for efficient communication with lower overhead at coarser levels of MGCG (conjugate gradient solver with multigrid preconditioner) solvers. One direction is a development of new algorithms with reduced communications, where trade-off between efficiency and robustness is very critical. More practical approach is aggregating MPI processes at coarser levels. In post-peta/exa-scale systems, each node will consists O(10^2) of cores, therefore utilization of these many cores on each node should be considered [3].

2.3 ppOpen-AT

Automatic tuning (AT) enables smooth and easy shift to further development on new/future architectures through ppOpen-AT/STATIC and ppOpen-AT/DYNAMIC. Directive-based special AT language for specific procedures in scientific computing, focused on optimum memory access, is developed. Features of ppOpen-AT are as follows:

- ppOpen-AT automatically and adaptively generates optimum implementation for efficient memory accesses in processes of methods for scientific computing in each component of ppOpen-APPL, such as explicit time marching procedures, matrix assembling procedures, and implicit linear solvers, under various types of environments (architecture of supercomputer system, available resources, problem size).
• ppOpen-AT also optimizes efficient memory accesses in main kernels of ppOpen-MATH.
• ppOpen-AT also optimizes load-balancing between host-CPU and co-processors/accelerators on heterogeneous computing nodes.
• ppOpen-AT automatically generates optimized codes written in CUDA or OpenCL for accelerators.

All of five types of methods for scientific computing in ppOpen-APPL are memory-bound application, if we consider FMM type approach for BEM. Therefore, we are focusing on optimization of memory accesses in various types of procedures of scientific applications.

ppOpen-AT will also optimize widely-used open-source applications and numerical libraries, such as OpenFOAM and PETSc.

2.4 ppOpen-SYS

ppOpen-SYS consists of libraries related to node-to-node communication (ppOpen-SYS/COMM) and fault-tolerance (ppOpen-SYS/FT). ppOpen-SYS/COMM will enable direct communications between co-processors (e.g. Intel MIC).

3. Schedule for Public Release

Developed libraries of ppOpen-HPC will be open for public. ppOpen-HPC will be installed to various types of supercomputers, and will utilized for research and development using large-scale supercomputer systems. Moreover, ppOpen-HPC will be introduced to graduate and undergraduate classes of universities for development of parallel simulation codes for scientific computing.

• 3Q 2012
  ➢ ppOpen-HPC for Multicore Cluster (K computer, Cray etc.)
  ➢ Preliminary version of ppOpen-AT/STATIC
• 3Q 2013
  ➢ ppOpen-HPC for Multicore/GPU Cluster
• 3Q 2014
  ➢ Prototype of ppOpen-HPC for Post-Peta-Scale System
• 3Q 2015
  ➢ Final version of ppOpen-HPC for Post-Peta-Scale System
  ➢ Further optimization on the target system

References

Figure 4. Coupled Ocean-Atmospheric Simulation System through ppOpen-MATH/MP

Figure 5. Examples of Directives for AT provided by ppOpen-AT/STATIC, Loop Fusion/Split for Explicit FDM code

<table>
<thead>
<tr>
<th>LoopFusionSplit region start</th>
</tr>
</thead>
<tbody>
<tr>
<td>LoopFusionSplit region end</td>
</tr>
</tbody>
</table>

NICAM: Semi-Unstructured Grid
B sử dụng phân vùng
MIROC-A

COCO: Tri-Polar FDM

Regional Ocean Model

Semi-Nested Grid

Pre- and post-process
Fault tolerance

Post-Peta-Scale System

Ocean Model

Atmospheric Model-1

Atmospheric Model-2