



LAWRENCE  
LIVERMORE  
NATIONAL  
LABORATORY

# Performance Characterization of LLNL HPC Codes

A. Bhatele

May 6, 2013

## **Disclaimer**

---

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

# Performance Characterization of LLNL HPC Codes

Abhinav Bhatele

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory  
P. O. Box 808, Livermore, California 94550  
E-mail: [bhatele@llnl.gov](mailto:bhatele@llnl.gov)

## Abstract

This is a working document for collecting profiling information of various LLNL HPC codes.

## 1 Introduction

For each application, we describe the application simulation loop, domain decomposition, the computational workload and communication structure. We also provide some simulation examples and profiling data.

## 2 pF3D

pF3D [1, 2] is a multi-physics code used to study laser plasma-interactions in experiments conducted at the National Ignition Facility (NIF) at LLNL. It is used to understand the measurements of scattered light in NIF experiments and also to predict the amount of scattering in proposed designs.

### 2.1 pF3D Simulation Loop

### 2.2 Domain Decomposition

pF3D operates on a logical three-dimensional (3D) process grid whose  $Z$ -direction is aligned with the laser beam. Let us suppose that the global simulation has  $nxtot \times nytot \times nztot$  zones and each process owns a domain of size  $nxloc \times nyloc \times nzloc$  zones.

$$nxtot = p_x \times nxloc \tag{1}$$

$$nytot = p_y \times nyloc \tag{2}$$

$$nztot = p_z \times nzloc \tag{3}$$

$p_x$ ,  $p_y$  and  $p_z$  are the number of processes in the  $X$ ,  $Y$  and  $Z$  direction respectively in the logical 3D grid. It should be noted that the mapping of this logical grid to the actual cores and interconnect topology is machine-dependent.

## 2.3 Computational Workload

## 2.4 Communication Structure

The simulation has three distinct communication patterns that are used in: the transverse derivatives in Maxwell’s equations (hereafter called the transverse wave equation), light advection, and solving the hydrodynamic equations. The transverse derivatives are handled using a spectral method, so they involve two-dimensional (2D) Fast Fourier Transforms (FFTs) in planes orthogonal to the laser, i.e. the  $XY$ -planes. More specifically, they are solved by gathering complete lines in the  $X$  and  $Y$  directions into the memory of a single process using `MPI_Alltoall` calls and then performing ordinary one-dimensional FFTs.

The light advection pattern consists of exchanging planes using `MPI_Send` and `MPI_Recv` calls between adjacent domains in the  $Z$ -direction. The sixth order advection option requires the exchange of up to three planes with the neighboring domains.

The hydrodynamic pattern consists of nearest-neighbor exchanges of the data on domain faces in the positive and negative  $X$ ,  $Y$  and  $Z$  directions (using `MPI_Isend`’s and `MPI_Recv`’s). This pattern is often referred to as a “halo exchange”. The hydrodynamic equations are typically solved after every 50 light propagation steps and take much less time than either of the other two patterns.

Given this structure, pF3D’s natural domain decomposition is to have  $p_z$  “slabs” (several adjacent  $XY$  planes) which are split further into  $p_x$  rows and  $p_y$  columns resulting in  $p_x \times p_y \times p_z$  domains each assigned to an MPI process. Within each slab, rows and columns are arranged into sub-communicators for the all-to-all’s discussed above. In particular, for weak scaling the mesh is extended along the  $Z$ -direction, adding more  $XY$  slabs and thus using more processors.

### 2.4.1 Transverse Wave Equation

Let’s pretend that the `MPI_Alltoall` is implemented using point-to-point messages. Each process “owns”  $nxloc \times nyloc$  zones within a  $Z$ -plane. When pF3D passes messages in the  $X$ -direction, those  $nxloc \times nyloc$  zones are split evenly between the  $p_x$  processes in that row of the domain decomposition. A process with logical coordinates  $(x, y, z)$  sends messages to  $p_x$  processors in its row with coordinates  $(*, y, z)$ . The size of each message in bytes is:

$$msg_x = \text{sizeof}(\text{complex}) \times nxloc \times nyloc / p_x$$

For the  $Y$ -direction all-to-alls, a process with coordinates  $(x, y, z)$  sends messages to  $p_y$  other processes in its column with coordinates  $(x, *, z)$ . The equivalent sizes when passing messages in the  $Y$ -direction are:

$$msg_y = \text{sizeof}(\text{complex}) \times nxloc \times nyloc / p_y$$

### 2.4.2 Light Advection

In the light advection pattern, a process in an  $XY$  domain with coordinates  $(x, y, z)$  communicates with corresponding processes in the two adjacent  $XY$  domains,  $(x, y, z - 1)$  and  $(x, y, z + 1)$ . Three planes are passed in the downstream direction and two in the upstream direction. These messages are larger than FFT messages because they are sent to nearest neighbor domains, not to all the domains in a row or column. The size of the larger of the two messages (in bytes) is:

$$msg_{adv} = 3 \times \text{sizeof}(\text{complex}) \times nxloc \times nyloc$$

### 2.4.3 Hydrodynamic Exchange

For the hydrodynamic calculations, every process communicates with its six neighbors in 3D similar to a seven-point halo exchange. A process with logical coordinates  $(x, y, z)$  communicates with  $(\pm x, y, z)$ ,  $(x, \pm y, z)$  and  $(x, y, \pm z)$ . Each message consists of one plane of real numbers:

$$msg_{hydro} = sizeof(real) \times nxloc \times nyloc$$

## 2.5 Simulation Examples

Table 1 shows the number of bytes exchanged between pairs of processes for pF3D simulations that have been run on the Blue Gene/L system at Lawrence Livermore National Laboratory (LLNL) and Cielo, a Cray XE6 at Los Alamos National Laboratory. It also shows three simulations that are candidates to be run on Sequoia, a Blue Gene/Q system at LLNL. The three Sequoia simulations have the same total number of zones and will use the same number of cores. The first one will use four MPI processes per core, the second will use two processes per core, and the third will use one process per core.

System	$p_x$	$p_y$	$p_z$	$nxloc$	$nyloc$	$nzloc$	$msg_x$	$msg_y$
Blue Gene/L	32	32	192	96	64	18	1536	1536
Sequoia	128	128	256	128	128	16	1024	1024
Sequoia	64	128	256	256	128	16	4096	2048
Sequoia	64	64	256	256	256	16	8192	8192
Cielo	16	16	128	640	192	34	61440	61440
Cielo	16	32	128	1280	160	34	102400	51200

Table 1: Message sizes (in bytes) for the transverse wave equation phase of pF3D (1D FFTs)

## 2.6 Profiling Data

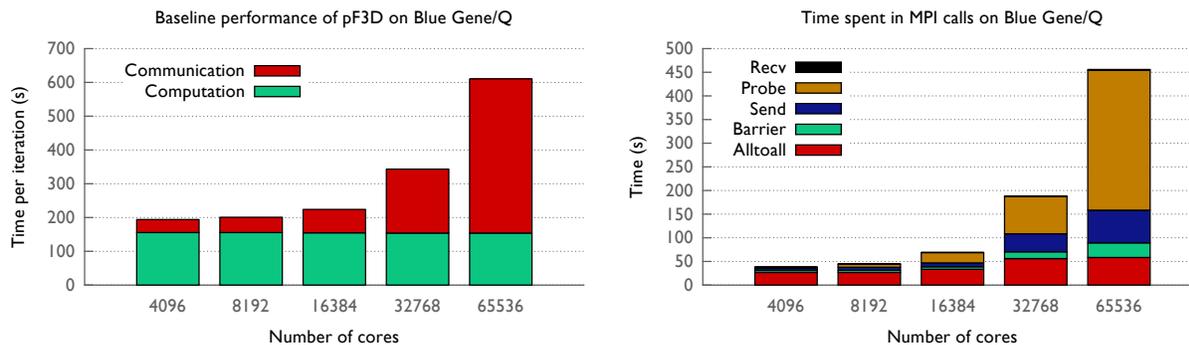


Figure 1: Profiling of pF3D on Blue Gene/Q (Mira)

## 3 Qbox

Qbox is a first-principles molecular dynamics (FPMD) code used to study the electronic structure of molecular systems to understand their material properties. The method uses a quantum description for electrons which is computationally intensive. Unlike classical molecular dynamics, FPMD does not have a simple domain decomposition that restricts communication to nearby processors, but instead necessitates frequent non-local communication. Many problems of interest require efficient strong scaling to hundreds of thousands of processors, where communication costs can be significant or even prohibitive.

### 3.1 Qbox (FPMD) Simulation Loop

Each molecular dynamics step requires a self-consistent iterative convergence to the electronic ground state, with the following steps:

```
for iscf = 1, nscf
  compute charge density
  compute xc potential
  for ite = 1, nite
    compute total energy
    apply Hamiltonian operator to wavefunction, update wavefunction
    Gram-Schmidt orthogonalization
  end
  subspace diagonalization
end
```

The inner iteration loop typically uses only 2-4 steps, whereas the outer loop can use anywhere from 3-100 steps, depending on the starting point and the smoothness of the potentials.

### 3.2 Domain Decomposition

Qbox uses a 2D process grid for its data distribution and communication. The main data object is the electronic wavefunction ( $ngw \times nst$ ),  $nst$  electronic orbitals each expanded in a plane wave basis of  $ngw$  complex coefficients. The wavefunction is distributed in a column-major distribution over  $nprow \times npcold$  MPI tasks. Each process column owns  $nst/npcol$  orbitals, distributed so that any given process row has the same subset of basis functions ( $ngw/nprow$ ) of all orbitals.

### 3.3 Computational Workload

The computational workload is primarily a mix of 3D FFTs and dense linear algebra, and is described in detail in the next section. Each step in the iteration loop consists of a mix of computation and communication, as follows:

#### 3.3.1 Compute Charge Density

The electronic orbitals on each process column are transformed to real space with a 3D FFT, then summed across process rows using `MPI.Allreduce` to compute the total charge density. A copy of the charge density is distributed over each process column.

### 3.3.2 Compute XC Potential

The charge density gradient is computed on each process column using a 3D FFT, and the exchange-correlation potential is computed locally from the density and gradient points stored on each task in the process column using one of several density functionals.

### 3.3.3 Compute Total Energy

The local contribution to the kinetic energy is computed, and then summed over all tasks (1 double, Allreduce sum over all tasks). Other energy terms are computed from local data using loops and zgemms and summed over all tasks within each process column (1 double, Allreduce sum over *nproc* tasks).

### 3.3.4 Apply Hamiltonian Operator, Update Wavefunction

The Hamiltonian operator is applied to local wavefunction data using loops and zgemms, followed by two 3D FFTs (forward and backward) for each orbital. The resulting data structure has the same size and data layout as the original wavefunction, which is then used to update the wavefunction with a parallel zgemm over all tasks.

### 3.3.5 Gram-Schmidt Orthogonalization

The new wavefunction is orthogonalized using Gram-Schmidt: compute  $S = Y^T Y$  using a parallel rank-k update (pzherk), perform a Cholesky decomposition on  $S$  (pzpotrf), followed by a triangular solve (pztrsm).

### 3.3.6 Subspace Diagonalization

The product of the transposed wavefunction  $Y^T$  and the Hamiltonian operator  $HY$  are computed with a parallel matrix multiply, then a symmetric eigensolve is performed to compute the eigenvalues and eigenvectors.

## 3.4 Communication Structure

3D FFTs are done within each column of the 2D grid. Planes of data are distributed over tasks in the process column. In the forward direction, FFTs in x- and y- are done locally with loops of 1D FFTs, followed by an Alltoallv transpose (over *nproc* tasks), followed by the remaining 1D FFTs in the z-direction. Backward transforms do the same operations in reverse order.

Parallel matrix multiplication is carried out over all tasks using ScaLAPACK, which manages the communication. Any threading is handled by the use of threaded single-node kernels.

Gram-Schmidt and subspace diagonalization both involve matrix multiplication of the form  $A^T A$ , the product of which ends up distributed on just the upper  $n_{pcol} \times n_{pcol}$  square of the process grid. In cases where the process grid is very rectangular ( $n_{prow} \gg n_{pcol}$ ), this may lead to a very small fraction of tasks carrying out the parallel Cholesky decomposition and symmetric eigensolve, both of which are ScaLAPACK calls.

## 3.5 Simulation Examples

To give an idea of typical data sizes and process grid dimensions, timings of a recent simulation are presented. The system is a carbon electrolyte, with 192 carbon atoms, 192 oxygen atoms, and 256

hydrogen atoms (2176 total electrons). The plane wave basis is defined by an energy cutoff of 85 Ry, and the electronic wavefunction matrix has dimension  $322048 \times 1216$ . The charge density uses a higher resolution basis of 2465104 plane waves, a copy of which distributed across each process column.

Running on 128 nodes of the LLNL Cab machine with 16 MPI tasks per node and one thread (2048 MPI tasks total), the optimal process grid was found to be  $256 \times 8$ . Each MD iteration had 5 self-consistent iterations with two inner iterations. The time per MD iteration was 38 seconds on average, with the following breakdown:

Phase	Time (s)
compute charge density	11.2
compute xc potential	0.2
compute total energy	2.1
apply Hamil., update wf	14.5
Gram-Schmidt orthog	6.0
Subspace diagonalization	5.7

Table 2: Time for different phases in Qbox

The workload is spread pretty evenly across 3D FFTs, parallel matrix multiplication and the square dense linear algebra operations. The fraction of time spent in communication was not directly measured for this run, but is typically 20-30% for a simulation of this size.

### 3.6 Profiling Data

## 4 Conclusion

## Acknowledgments

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This work was funded by the Laboratory Directed Research and Development Program at LLNL under project tracking code 13-ERD-055 (LLNL-TR-XXXXXX).

## References

- [1] R. L. Berger, B. F. Lasinski, A. B. Langdon, T. B. Kaiser, B. B. Afeyan, B. I. Cohen, C. H. Still, and E. A. Williams. Influence of spatial and temporal laser beam smoothing on stimulated brillouin scattering in filamentary laser light. *Phys. Rev. Lett.*, 75(6):1078–1081, Aug 1995.
- [2] C. H. Still, R. L. Berger, A. B. Langdon, D. E. Hinkel, L. J. Suter, and E. A. Williams. Filamentation and forward brillouin scatter of entire smoothed and aberrated laser beams. *Physics of Plasmas*, 7(5):2023, 2000.