MPI/OpenMP Parallelization of the Fragment Molecular Orbital Method in GAMESS

Abstract
In this work, we present a novel parallelization strategy for the Fragment Molecular Orbital (FMO) method in the quantum chemistry package GAMESS. The original FMO code has been parallelized only with MPI, which limits scalability of the code on multi-core parallel computing. To address this problem, we parallelize FMO with a new hybrid MPI+OpenMP scheme that shows excellent scaling up to 2,048 Intel Xeon Phi nodes (337,672 cores) on Theta supercomputer. MPI-OpenMP code not only scales well compared to MPI code, but also performs up to two times faster and has significantly smaller memory footprint.

Fragment Molecular Orbital (FMO) Method
In FMO, a molecular system is divided into fragments and the total properties, such as the energy or its gradient, is calculated from those fragments and (in FMO) their pairs, computed in the embedding potential. FMO treats the electrostatic and exchange repulsion between the fragments. It is widely used due to its low computational cost and high performance.

GAMESS (General Atomic and Molecular Electronic Structure System)
GAMESS is a quantum chemical package that handles a variety of problems, from Hartree-Fock theory to highly correlated methods. GAMESS is widely used in the scientific community due to its high performance and flexibility.

FMO algorithm in GAMESS
- **Basic steps of FMO (FMO2) calculation:**
  1. Compute one-electron density distribution of each fragment.
  2. Construct initial guess wavefunction of each fragment.
  3. Iterate self-consistent field calculations.
  4. Diagonalize the wavefunction of fragments in a field of other fragments.
  5. The latter is described by the embedding electronic potential (ESP). This step stops until all fragments converge.

Goals of the study
- Eliminate or reduce drawbacks of MPI-only parallelization:
  1. Excessive memory utilization on massively parallel architectures.
  2. Only half of GAMESS processes are doing computation, others are used as data-converters.

Solution
- Implement multithreaded algorithms for intensified parallelization of FMO-specific parts of code.

MPI+OpenMP parallelization of one-electron integrals calculation
- **Step 1:** parallelize the integrals calculation with MPI.
- **Step 2:** parallelize the inter-fragment communication with OpenMP.

Conclusions
In this work, the conversion of the MPI-only GAMESS FMO code to hybrid MPI+OpenMP version is described. The resulting hybrid implementations are benchmarked on the same supercomputers and computational nodes for scalability. The new hybrid MPI+OpenMP code significantly improves the original MPI-only code in GAMESS. It scales badly up to 2,048 and quickly becomes twice faster than MPI-only version.

Benchmark results
- **Figure 3:** Comparison of the new MPI+OpenMP and the original MPI-only GAMESS. Time in seconds compared to the whole FMO calculation on Theta supercomputer.