

## 不均一クラスタ上での並列 Linpack の性能に関する検討

岸本芳典<sup>†</sup> 市川周一<sup>†</sup>

HPL (High Performance Linpack) は分散メモリ型並列計算機用の Linpack である。HPL を不均一クラスタ上で実行すると負荷の不均衡により性能が低下するが、この問題は高速 PE 上で複数のプロセスを起動することにより改善できる。高速 PE 上では複数プロセスの起動により性能オーバーヘッドが生じるが、実測例では性能ロスは 3 割以下であった ( $N \geq 7000$ )。このときクラスタ全体の性能は、負荷不均衡が解消されることにより最大 3 割程度改善された。本研究では (1) 高速 PE 上のプロセス数, (2) クラスタの PE 構成, (3) ネットワーク構成などを変えて性能を測定した。しかし最善の実行方法は問題サイズ  $N$  やクラスタ構成によって異なるので、今後は自動的に最適な実行条件を選ぶ方法を検討する。

### Parallel Linpack Benchmark Results on Heterogeneous Cluster

YOSHINORI KISHIMOTO<sup>†</sup> and SHUICHI ICHIKAWA<sup>†</sup>

HPL (High Performance Linpack) is a benchmark for distributed memory parallel computer. However, the performance of HPL is degraded by load unbalance on a heterogeneous cluster. This problem can be improved by running multiple processes on fast processing elements (PE), though the performance overhead emerges on fast PE by running multiple processes. According to our measurements, the performance loss of this method was less than 30 percent ( $N \geq 7000$ ), while the total performance of a heterogeneous cluster was improved about 30 percent by dissolving load unbalance. This paper also shows (1) the performance for various number of processes on fast PE, (2) the performance for various PE configuration, and (3) the effect of network speed. It is left for future work to find the best execution method and configuration for the given size of the problem.

#### 1. はじめに

多くの科学技術計算応用では、行列などの線型代数演算の高速化は極めて重要である。線型代数演算のうち LU 分解の性能測定には、Linpack ベンチマークが広く用いられている。特にクラスタなどの分散メモリ型並列計算機には、MPI (Message Passing Interface) を用いて並列化した HPL (High Performance Linpack Benchmark)<sup>1)</sup> が多く利用される。

クラスタは高速ネットワークで要素計算機 (PE) を結合した仮想の並列計算機で、システム要素が均一の PE による構成が一般的である。HPL はこれを前提として各 PE に負荷を均等に分割するが、不均一な PE を持つクラスタでは負荷の不均衡によって性能上の問題を生ずる。

しかし、システム要素を統一する必要が無ければ、既存クラスタを最新のプロセッサで増強することや、手持ちの計算機を組み合わせたクラスタを構築するこ

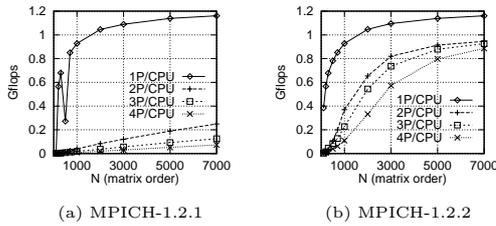
とが可能となる。このため、不均一クラスタの対費用効果は均一な場合よりもさらに向上すると期待できる。

図 3(a) は、不均一なクラスタ上で HPL を実行した結果である。HPL ( $N=10000$ ) で各 PE の単体性能を測定した場合、Athlon 1.33GHz (1.15 Gflops) は PentiumII 400MHz (0.27 Gflops) の約 4 倍高速となる。ところが、Athlon  $\times$  1 + PentiumII  $\times$  4 の性能は PentiumII  $\times$  5 の性能に等しい。これは、高速な Athlon は演算を先に終了するが、アイドル状態になって PentiumII の処理の終了を待つためである。

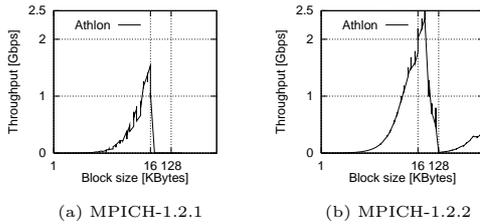
#### 2. マルチプロセスによる負荷分散

この問題を避けるには各 PE の速度に応じた負荷を与えればよい。笹生ら<sup>2)</sup> は PE の速度に応じた計算負荷を処理できるように HPL を修正したが、その方法では応用 (HPL) や実装に依存してしまう。一方、高速な PE に複数の MPI プロセスを起動する方法ならば、均一クラスタ用に書かれた各種の並列応用にも容易に適用できる。ただし同一 PE 上に複数プロセスを起動するとオーバーヘッドで性能が低下するため、笹生らは性能低下が著しいとして本手法を棄却している<sup>2)</sup>。

<sup>†</sup> 豊橋技術科学大学 知識情報工学系  
Department of Knowledge-based Information Engineering,  
Toyohashi University of Technology



(a) MPICH-1.2.1 (b) MPICH-1.2.2  
 図 1 MPICH のバージョンによる演算性能の差



(a) MPICH-1.2.1 (b) MPICH-1.2.2  
 図 2 MPICH のバージョンによる通信性能の差

図 1 に、Athlon 上に 1~4 プロセスを起動した場合の性能を示す。MPI の実装としては MPICH を選んだ。MPICH-1.2.1 では複数プロセス起動によって 8~9 割近い性能低下が見られるが、1.2.2 では 2~3 割の低下にとどまっており実用に耐え得る性能を示す。

MPICH-1.2.1 での性能低下にはオーバーヘッド以外の要因が存在する。図 2 は、同一 Athlon 上に起動した 2 プロセス間の通信性能を NetPIPE を用いて測定した結果である。MPICH-1.2.1 ではブロックサイズ 16KB 以降で性能が極端に悪化し、ほぼ 0 となっている。一方、MPICH-1.2.2 では 128KB 近辺での一時的な性能低下にとどまり、それ以上のサイズでは回復している。

以上見てきたように、複数プロセス起動によるオーバーヘッドは存在するが、本手法の適用範囲の広さを考えれば許容可能な割合であると思われる。

### 3. 最適な実行方法とは

不均一クラスタ上で HPL を実行する場合、実現可能な実行方法の中から最良の選択肢を発見する必要がある。まず、高速な PE のみを使用するか、低速な PE も使用するか、という選択が必要である。さらに複数プロセスを実行して負荷を均衡化する場合は、高速な PE 上でいくつプロセスを起動するのか、を決定しなければならない。

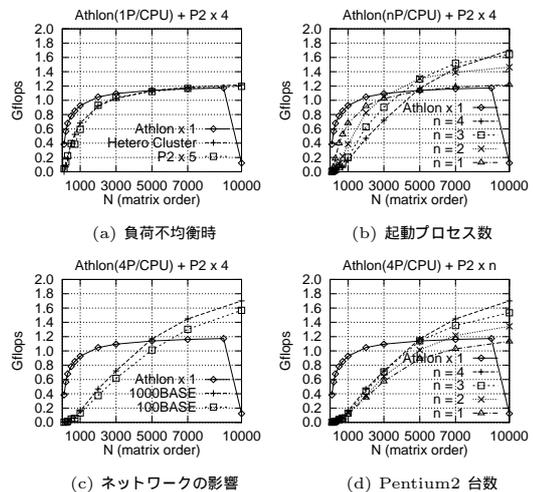
そこで、これらの選択肢が性能に及ぼす影響を調べた。測定は、Athlon 1.33GHz × 1 と PentiumII 400MHz × 4 を 1000baseSX で結合したクラスタで行った。

図 3(b) は、高速な Athlon 上のプロセス数と性能の関係である。行列のサイズ  $N$  によって最適なプロセス数は異なっている。例えば、 $N \leq 3000$  では Athlon 1

台だけが良く、 $3000 \leq N \leq 5000$  では PentiumII × 4 も利用し Athlon 上に 2 プロセス、 $5000 \leq N \leq 9000$  では同じく 3 プロセス起動するのが良い。

しかし、このような切り替え点 (サイズ  $N$ ) は一定でなく、ネットワークや計算機構成に依存する。例えばネットワークが 100baseTX なら切り替え点の  $N$  は大きくなり (図 3(c))、PentiumII の利用可能台数が 4 より小さい時も切り替え点の  $N$  は大きくなる (図 3(d))。

また、各 PE の搭載記憶容量により選択肢が制限される場合もある。負荷に対して主記憶が不足した場合にはスラッシングが発生し、クラスタ性能がかえって低下するためである。このため行列サイズが大きい場合には、PE 上で起動するプロセス数を少なくする、あるいは性能の低い PE の記憶も利用する、などの対策が必要となる。



(a) 負荷不均衡時 (b) 起動プロセス数  
 (c) ネットワークの影響 (d) Pentium2 台数  
 図 3 ヘテロクラスタの各パラメータにおける性能

### 4. おわりに

今後は、クラスタ性能を自動的に測定し、多くの選択肢から最適な実行環境を自動的に選ぶ手法の開発に取り組んでゆく。

### 参考文献

- Petit, A., Whaley, R. C., Dongarra, J. and Cleary, A.: HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers. <http://www.netlib.org/benchmark/hpl/>.
- 笹生健, 松岡聡, 建部修見: ヘテロなクラスタ環境における並列 LINPACK の最適化, 情処研報 2001-HPC-86, pp. 49-54 (2001).