

Linpack Benchmark のアルゴリズム解析
釘井 睦和

1 前回からの課題

- Linpack Benchmark のアルゴリズム解析
- ロードモニタ Ganglia のインストール
- Xenia の Linpack 計測

2 達成状況および研究報告

2.1 Linpack Benchmark のアルゴリズム解析

Linpack Benchmark(以下 Linpack) は, 米国テネシー大学の J.Dongarra 博士によって開発された LU 分解にもとづく連立一次方程式の解法プログラムである。もとは, 線形代数問題の LU 分解を行うためのサブルーチンの集合であり, ベンチマークにも利用されていると表現するほうが正確である。ベンチマークとしての Linpack は, 計算機で頻繁に利用する命令を集めたカーネルベンチマークであったが, 実行結果に分散メモリ型並列計算機のためのベンチマークとして HPL(High-Performance Linpack Benchmark) が開発された。Linpack を計測するために用いられる HPL の各種パラメータについて調べた。

問題サイズ N やブロックサイズ NB , プロセス格子 (P, Q) は HPL を実行する上で最も重要なパラメータだが, 本報告では通信に用いられる 6 つのトポロジーについて述べる。

以下にその 6 つの通信トポロジーを示す。

1. Increasing-ring
2. Increasing-ring (modified)
3. Increasing-2-ring
4. Increasing-2-ring (modified)
5. Long (bandwidth reducing)
6. Long (bandwidth reducing modified)

それぞれについて, さまざまなクラスタ環境で 16 プロセッサを用いて計測を行った。

Table 1 のパラメータで測定した結果を Table 2 に示す。MPI ライブラリは, cambria は lam を gregor は lam-gm を用いた。

Table 1 パラメータ

N	20000
NB	64
P, Q	4, 4

Table 2 16CPU での linpack 結果 (単位:GFlops)

	cambria	gregor	gregor(dual)
1	5.03	8.15	7.25
2	5.10	8.26	7.34
3	5.10	8.25	7.35
4	5.10	8.26	7.36
5	5.12	8.22	7.32
6	5.07	8.29	7.38

この結果より, プロセスグリッドをスクウェアにすると通信に用いるトポロジーは値に大きく影響はしないことがわかる。gregor に関して dual の方が遅くなっているのは, 1CPU 当たりのメモリ量が異なるためである。今後は, プロセスグリッドをスクウェアにしない場合の通信, NB の値と LU 分解の方法などの関係などについて調べる。

2.2 ロードモニタ Ganglia のインストール

Ganglia Cluster Toolkit(以下 ganglia) はクラスタにおけるさまざまなマシン状況をブラウザで表示させるツールである。この ganglia を 3 台のディスクフルクラスタに導入し, apache と PHP を用いてブラウザで表示させた。Fig. 1 に表示されている様子を示す。



Fig. 1 ロードモニタの表示結果

2.3 Xenia における Linpack 計測

進捗が遅れているため計測はまだ行われていない。

3 今後の課題

- ganglia のインストールマニュアルの作成
- ganglia のカスタマイズ
- Xenia の Linpack 計測