

Speeding Up NEC2++ by Multi-core Processor Yoshiyuki Takeyasu / JA6XKQ

マルチコア・プロセッサによる NEC2++ の高速化 武安義幸 / JA6XKQ

NEC2++ [1] の数値演算ライブラリを ATLAS [2] から OpenBLAS [3] に置換して、マルチコア・プロセッサ対応による高速化を図った。[4] マルチコアへの対応が簡単にできるようになると単純な欲望が頭をもたげ、プロセッサのコア数を更に増やして計算速度の高速化を試したくなった。

特に新たな知見はないが、自己満足の記録として残す。

機器構成

コンピュータのハードウェアとソフトウェアの構成は下記のとおり。

- プロセッサ : Intel Xeon X5570
- コア数 : 4
- クロック周波数 : 2.93 GHz
- プロセッサ数 : 2
- OS : Debian Ver. 7.0 Wheezy
- gcc : Ver. 4.7.2
- OpenBLAS : Ver. 2.5
- NEC2++ : Ver. 1.5.1

コンピュータの総コア数は 8 であり、従来環境 [4] の二倍となる。OpenBLAS と NEC2++ は、新たなソフトウェア環境下で再ビルドした。OpenBLAS は、プロセッサの Hyper Threading (HT) 機能を ON にすると実行速度が遅くなるので、HT 機能を OFF としている。

実行時間比較

3 セクション・コニカル・ホーンの計算モデル [5] の実行時間比較を下表に示す。

Model No. of surface- patch	3.16GHz Core2 Duo ATLAS (sec)	3.0GHz Core2 Quad OpenBLAS (sec)	2.93 GHz Dual X5570 OpenBLAS (sec)
6,319	360	135	68
9,968	1,380	510	253
13,992	3,600	1,360	669

考察

4 コアの Core2 Quad に対して 8 コアの Dual X5570 が、コア数に比例した高速化を果たしている。

NEC2++ を gcc でコンパイルする際に `-pg` オプションを付けると、実行時間のプロファイルの取得が可能である。NEC2++ を実行すると、計算結果の他にプロファイル結果の `gmon.out` も出力されるので、それを `gprof` で表示する。図-1 に表示されたプロファイルを示す。

図-1 に示す実行時間のプロファイルから、実行時間の 90% が OpenBLAS の `zgemv_kernel_n` で占められていることが判る。この `zgemv_kernel_n` の部分がコア数に反比例して短縮される。したがって、90% が 半分の 45% に短縮され、コア数とは無関係な 10% と合わせて 55% の時間比で実行時間が短縮される。100% が 55% となるので、1.818 倍の高速化となる。

数値例を示すと、Core2 Quad での 135 秒が、 $135 \times 0.55 = 74$ 秒となる。この数値は両プロセッサのクロック周波数を同一と仮定しているため、両者のクロック周波数で換算する。X5570 にはクロックの Turbo Boost 機能があり、2.93 GHz のクロック周波数が最大で 3.333 GHz 相当となる。クロック周波数比において、Core2 Quad の 3.0GHz に対して 1.111 倍の高速化が見込まれる。コア数比とク

ロック比の総合では、 $1.818 \times 1.111 = 2.02$ 倍の高速化である。

上述の数値例を確認すると、 $136 \times 1/2.02 = 67$ 秒となり、実測値と推定値がほぼ一致している。

まとめ

プロセッサのコア数を 4 コアから 8 コアに増やし、約 2 倍の高速化が得られた。この高速性能を活かして、ホーン・アンテナの最適化設計に着手した。計算の加速により、最適化設計の検討も加速したい。

//
☆

Flat profile:

Each sample counts as 0.01 seconds.

%	cumulative	self	self	total		
time	seconds	seconds	calls	s/call	s/call	name
89.99	289.89	289.89				zgemv_kernel_n
3.25	300.37	10.48				sched_yield
1.62	305.60	5.23				ztrsm_kernel_LT
1.45	310.26	4.66				inner_advanced_thread
0.73	312.61	2.35	39929761	0.00	0.00	nec_context::hintg(double, double, double)
...						

図-1 : 実行時間のプロファイル (一部分のみ)

参考文献

- [1] NEC2++
Timothy Molteno
<http://elec.otago.ac.nz/w/index.php/Necpp>
- [2] Automatically Tuned Linear Algebra Software (ATLAS)
<http://math-atlas.sourceforge.net/>
- [3] OpenBLAS
<http://xianyi.github.com/OpenBLAS/>
- [4] Speeding Up NEC2++ by OpenBLAS
OpenBLAS による NEC2++ の高速化
武安義幸、JA6XKQ
http://www.terra.dti.ne.jp/~takeyasu/Nec2ppOpenBlas_1.pdf
- [5] Simulation of a horn antenna using NEC2++
NEC2++ によるホーン・アンテナのシミュレーション
武安義幸、JA6XKQ
<http://www.terra.dti.ne.jp/~takeyasu/Nec2pp3SecHorn.pdf>