

自動チューニング型の固有値ソルバーについて

直野 健
(株)日立製作所中央研究所
今村 俊幸

日本原子力研究所、High Performance Computing Center Stuttgart (HLRS)

要旨

実対称密行列の固有値計算ライブラリの自動チューニング方式を開発中である。本報告では、アンローリングの選択方式を詳細に検討することを目的に、日立 SR8000、IBM RS/6000 SP 上で、固有値のリダクション演算部分に対する複数のアンローリングを対象に、1 次元毎の性能測定、およびループのメモリ要求性能 Byte/Flop 値と実効性能のトレンド評価を行った。その結果、低段アンロールでは見られなかった性能劣化が、高段アンロールにおいて同じ次元で見られたこと、最適なアンロール選択は複数の次元において評価すべきことが確認できた。

An Evaluation towards an Automatic Tuning Eigensolver

Ken Naono
Hitachi Ltd., Central Research Laboratory
Toshiyuki Imamura
Japan Atomic Energy Research Institute/High Performance Computing Center
Stuttgart (HLRS)

Abstract

We are investigating a tuning method for an eigensolver for a dense symmetric matrix. In this paper, we discuss how to select the unrolling depth. For that, we evaluate the performance of various unrolled reduction loops of the eigensolver for every dimension on a Hitachi SR8000 and on an IBM RS/6000 SP. We also analyze the trend between the Byte/Flop and the performance for each case. We confirm that the performance is degraded with higher depth of unrolling in some dimensions where that is not occurred with lower depth of unrolling. We also confirm that selection of the unrolling depth should be examined on several dimensions.

1. はじめに

近年、スーパーコンピュータ単体での性能を引き出すライブラリだけではなく、産総研の Ninf[1]、テネシー大の NetSolve[2] などネットワークを介して手元のワークステーション・PC から、どこかのスーパーコンピュータかを意識せずに利用可能なネットワークライブラリが研究されてきた。この流れから、現在 Grid と呼ばれる、世界の多数の計算機を接続して仮想的な巨大計算を行う利用環境の研究に至っている。また 90 年代後半から、どの計算機上でもより高い性能を引き出す、自動チューニング機能を有したライブラリの研究が行われるようになった。LAPACK を自動チューニングするテネシー大の ATLAS[3]、並列ライブラリを自動チューニングする東大の I-LIB[4] などの研究プロジェクトがある。

今後、自動チューニング技術によって、ライブラリ開発者はユーザアクセスが想定される計算サーバ毎にチューニングされた高性能なライブラリを配布可能になり、一方でネットワークライブラリ技術によって、ユーザは、どこかのサーバに接続するかを意識せずにインターネットに接続するだけでライブラリ機能を利用できる、即ち GRID 化への道が開けると思われる。

ところが、GRID 環境下でのライブラリを利用する場合、性能の安定性に関する議論が重要であるが、既存の性能評価では安定性も含んだ詳細な評価がなされていなかった。具体的には、いくつかの次元においてメモリのバンクコンフリクトやキャッシュのカラムコンフリクト等による性能劣化が懸念されるが、その度合いが評価されていなかった。

そこで、本研究では、複数のアンローリングに対して 1 次元毎の詳細な性能評価を行う。また、特に固有値の計算では性能と Byte/Flop との関係性が深いので[5]、そのトレンドも評価するこ

とによって、最適なアンローリング段数の定め方について議論する。

2. 固有値計算アルゴリズムとアンローリングによるチューニング方式

固有値計算のアルゴリズムは、主に SR8000 向けライブラリ MATRIX/MPP の固有値計算ライブラリに実装した手法である、ブロック化されたハウスホルダ変換にループのアンローリングで高速化したもの[5]をベースとする。

チューニングは、リダクションループのアンローリングであり、アンローリングの段数を変更し、実際に性能を測定しつつ最適な組合せを採用する。

性能評価では、行列のサイズによって性能が変化するバンクコンフリクトの問題や次元数が小さい箇所ではオーバーヘッドのため実効性能が低く評価にそぐわない、などの問題がある。そこで、経験的に性能が比較的安定する 3000-4000 次元で 1 次元毎に評価を行い、それらの平均値をアンロールの結果の性能とみなす。実際の自動チューニング型ライブラリでは、1 次元毎の性能評価でアンロールの組を定めることは時間コスト上不可能であるが、本報告では、最も高精度なアンロール組の定め方を見た結果から、どの程度まで簡略化ができるのかを探ることとする。

アンロールの対象となるリダクション演算は、

```
do M = 1, MB
  do J = 1, N
    do I = 1, J
      A(I, J) = A(I, J) - ( V(I, M)*U(J, M) + U(I, M)*V(J, M) )
    enddo
  enddo
enddo
```

である。この三重ループに対するアンロールの組みは、外側から段数を表記することとする。例えば、(4-3-2)は、上記 M のループを 4 段、J のループを 3 段、I のループ 2 段とアンロールする展開を表す。

アンロール段数の組合せは、SR8000 では

```
(1-1-1), (1-1-2),
(2-2-1), (2-2-2),
(3-3-1), (3-3-2),
(4-4-1), (4-4-2),
(5-5-1), (5-5-2), (5-5-4),
(6-6-1), (6-6-2)
```

の 11 種類、RS/6000 SP では

```
(1-1-1),
(2-2-1), (2-2-2),
(2-3-2), (3-2-2),
(3-3-1), (3-3-2),
(4-4-1)
```

の 8 種類とする。なお、これらの組合せは理論上必要となるレジスタ数がハードウェアの有するレジスタ数の範囲に納まるものの中から選択したものである。例として、SR8000 における (2-2-2) のソースコードを図 1 に示す。この例のように、外側 2 段の展開は実際にコーディングを行い、最内側では `*soption unroll (2)` という指示文によってアンローリングを行う。元のループは上三角のみの計算であるが、中間ループのアンロールによって下三角部分を余分に更新することになる。これは計算全体から見た場合無視できる程度のものである。

```

        subroutine tunecore222(
&      i$1, i$5, i$3, m0_mod6, m0, i$nnod, i$inod, l1$a, u, v, nm
&      )
C
      implicit double precision (a-h,o-z), integer(i-n)
      double precision l1$a(nm,*), u(nm,*), v(nm,*)
C
      *voption vec
      *soption noloopreroll
      do i$1=i$5, i$3, 2
      *soption noloopreroll
      do i0=m0_mod6+1, m0, 2
        j$0=(i$1-1+0)*i$nnod+i$inod
        j$1=j$0+i$nnod
      *
        u0_0 = u(j$0, i0+0)
        u1_0 = u(j$1, i0+0)
        u0_1 = u(j$0, i0+1)
        u1_1 = u(j$1, i0+1)
      *
        v0_0 = v(j$0, i0+0)
        v1_0 = v(j$1, i0+0)
        v0_1 = v(j$0, i0+1)
        v1_1 = v(j$1, i0+1)
      *soption unroll(2)
      *soption noloopreroll
      *voption vec
      do k=1, j$1
      *
        l1$a(k, i$1+0)=l1$a(k, i$1+0)
        1      - u0_0*v(k, i0+0) - v0_0*u(k, i0+0)
        1      - u0_1*v(k, i0+1) - v0_1*u(k, i0+1)
      *
        l1$a(k, i$1+1)=l1$a(k, i$1+1)
        1      - u1_0*v(k, i0+0) - v1_0*u(k, i0+0)
        1      - u1_1*v(k, i0+1) - v1_1*u(k, i0+1)
      *
      enddo
      enddo
      *-
      enddo
      return
      end

```

図 1 (2-2-2)展開のソースコード

3. 各アンローリングの評価

1次元毎の詳細な評価結果について、図2に(*-*)1;*=1, 2, 3, 4, 5, 6のSR8000/F1(375MHz)の結果を示す。(1-1-1)から(6-6-1)まで徐々に性能が高くなっていくが、ところどころの性能のぶれから、アンロール段数を上げててもほとんど性能が変わらない次元数もあり、次元数によっては性能が逆転するところも散見され、ピンポイントでアンロールの効果を評価すると危険であることが分かった。例えば3280次元では、(6-6-1)展開よりも(4-4-1)展開が高性能だが、3000から4000次元を見てみるとおおよそ(6-6-1)展開のほうが高性能となった。

また性能が急激に落ちる部分もあった。(1-1-1)展開と(2-2-1)展開は性能が落ちなかったが、性能が1割程度も劣化する場合があった。例えば(6-6-1)展開では、他の次元では1110Mflop/s前後なのに対し、3057次元から3080次元までの間が1020Mflop/s前後となった。つまり、低段アンロールでは見られなかった性能劣化が、高段アンロールにおいて同じ次元で見られたことになる。また、このことから複数のポイント、できれば十から十数ポイントでの性能データによってアンロール組を決定すべきであることが分かる。

同様に図3には最内側2段にしたSR8000の結果を、図4にはRS/6000 SP(Power3, 375MHz)の結果を示す。上記と同様に(1-1-1)展開では性能劣化がない次元で、アンロール段数を上げていくと性能劣化を引き起こす次元が散見された。なお、3000-4000次元は、RS/6000 SPではL2キャッシュを越える計算サイズのため、1000次元前後の性能(約830Mflop/s)よりはやや低い状態である。

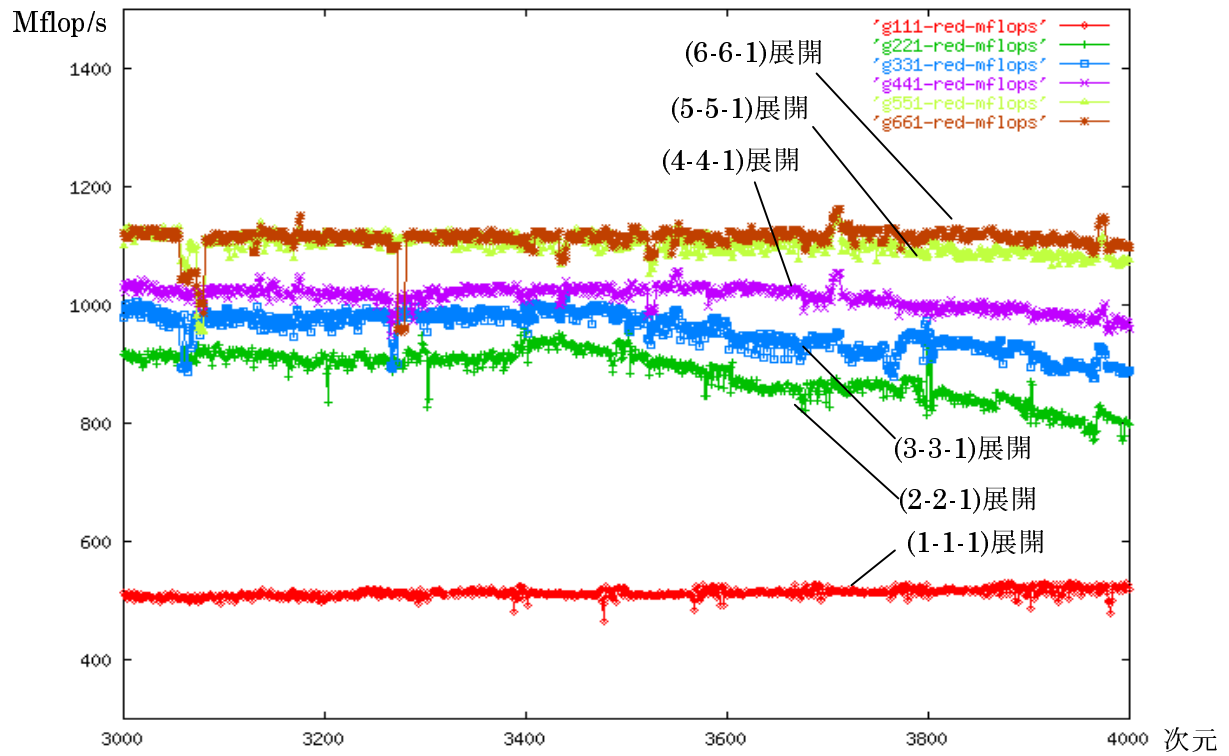


図2 1次元毎の詳細評価:最内1段 (SR8000, 1CPU, peak;1500Mflop/s)

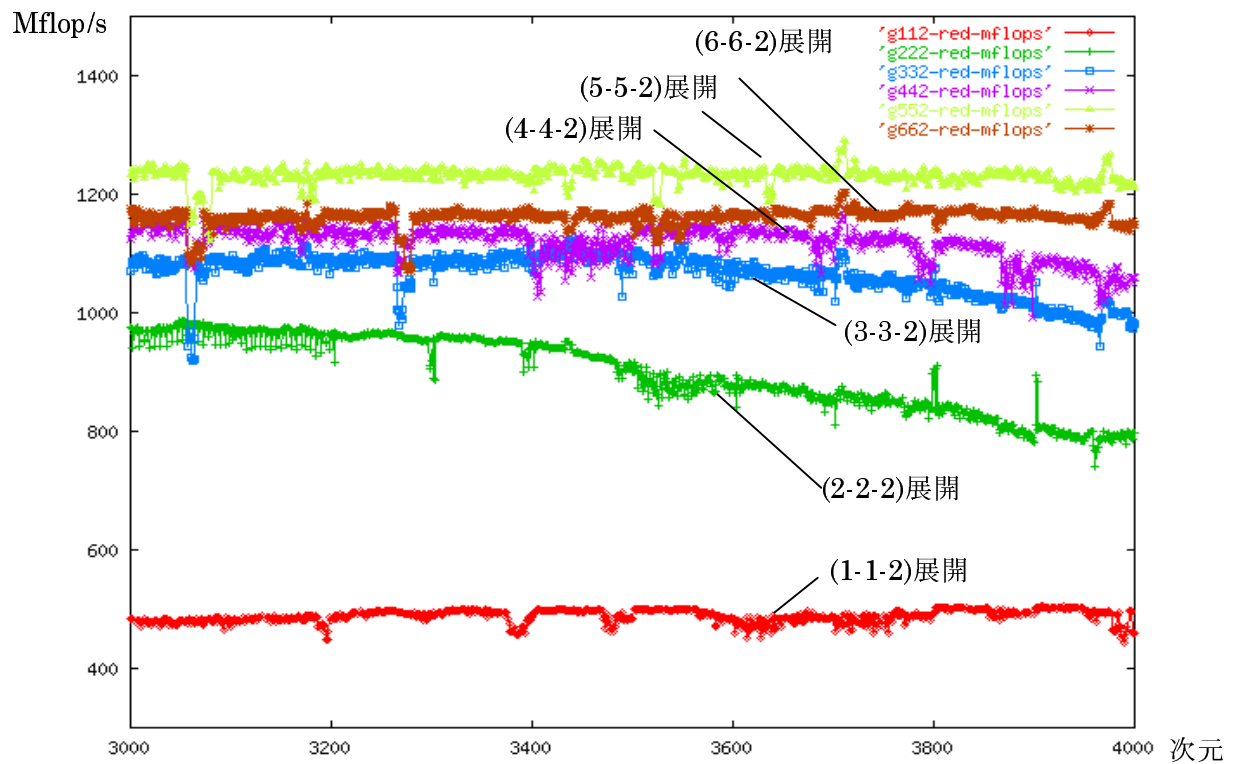


図3 1次元毎の詳細評価:最内2段 (SR8000, 1CPU, peak;1500Mflop/s)

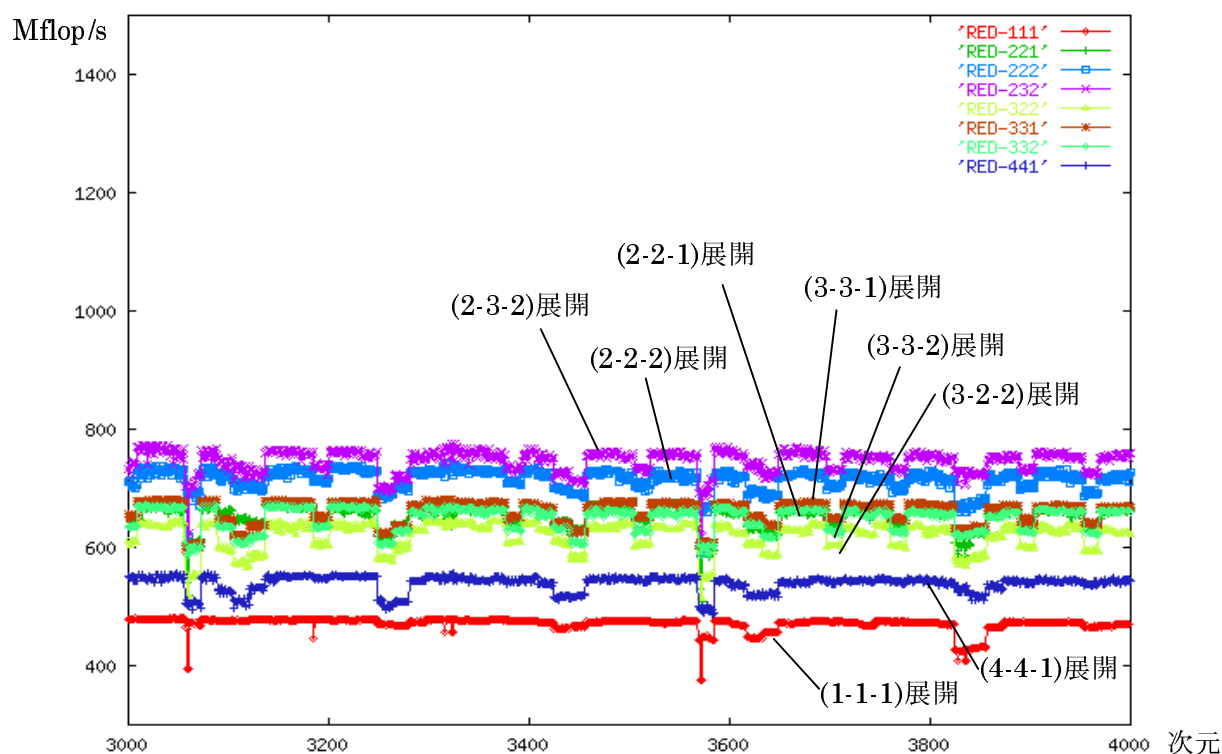


図 4 1次元毎の詳細評価 (RS/6000 SP, 1CPU, peak; 1500Mflop/s)

また、アンロール段数毎の Byte/Flop と 3000 次元から 4000 次元の平均性能との関係性を評価した。SR8000、RS/6000 SP について、前節のアンロールの組みと Byte/Flop の関係を表 5 に、Byte/Flop と性能のトレンドを図 6 に示す。ここで、Byte/Flop とは、各ループ演算処理に対して定義される指標で、浮動小数点演算 1 回あたりに要求する（転送されるべき）データ容量である。図 6 の”SR-unroll1. d”は、 $(*-*-1)$; $*$ =1, 2, 3, 4, 5, 6 の変化であり、”SR-unroll2. d”は、 $(*-*-2)$; $*$ =1, 2, 3, 4, 5, 6 の変化である。(5-5-4)のみ SR-unroll4. d1 点で示してある。このアンロールと Byte/Flop の関係は、 $(X-Y-Z)$ 展開の場合、Byte 数が $8\text{Byte} \times (2X+2Y) \times Z$ 、Flop 数が $4 \times XYZ$ であるので、Byte/Flop は $4(X+Y)/XY$ となる。

また RS/6000 SP の結果については、”SP3-unroll1. d”では $(*-*-1)$; $*$ =1, 2, 3, 4 を、”SP3-unroll2. d”では $(2-2-2)$ 、 $(2-3-2)$ 、 $(3-2-2)$ 、 $(3-3-2)$ に関して示している。 $(2-3-2)$ 、 $(3-2-2)$ とともに約 3.3Byte/Flop であるが、上の点が $(2-3-2)$ 展開である。

表 5 アンロールの組みと Byte/Flop の関係

アンロール	(6-6-1) (6-6-2)	(5-5-1) (5-5-2) (5-5-4)	(4-4-1) (4-4-2)	(3-3-1) (3-3-2)	(2-3-2) (3-2-2)	(2-2-1) (2-2-2)	(1-1-1) (1-1-2)
Byte/Flop	1.33333	1.6	2.0	2.6666	3.33333	4.0	8.0

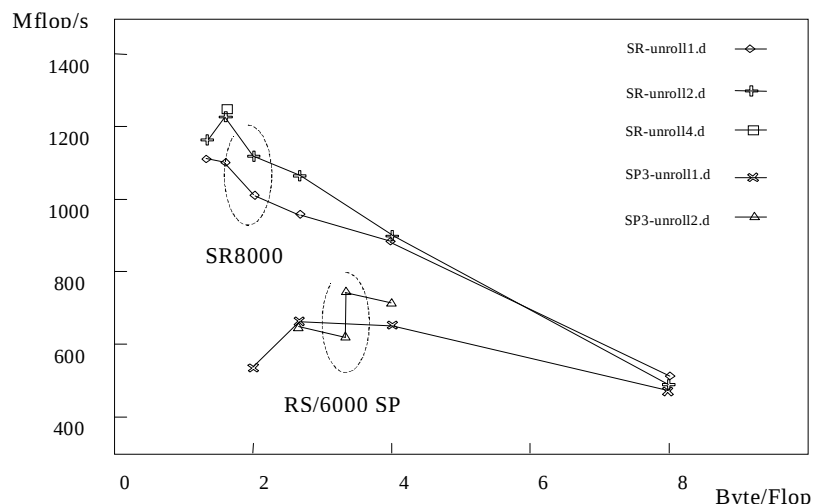


図6 Byte/Flop に対する実効性能トレンド

図6の結果から、SR8000もRS/6000SPもByte/Flopが徐々に小さくなるに従って性能が向上するものの、ある箇所から性能が下がってくる事が分かる。SR8000の場合、11種類の展開のうち(5-5-4)展開が最高の性能であり、(6-6-2)から性能が下がる結果となった。RS/6000SPでは8種類のうち、(2-3-2)展開が最高の性能であり、(3-2-2)展開や(3-3-2)展開では性能が劣化する結果となった。

SR8000もRS/6000SPも、3000次元から4000次元の間は、性能が約1割ほど劣化する次元があることも確認できたが概してほぼ一定した性能を発揮する部分であり、また、Byte/Flopと性能のトレンドを明らかにできたことから、この区間の平均をとって、アンロール段数の代表の性能とするのがアンロール段数選択のひとつの方法となり得る。

4. まとめ

本報告では、固有値計算ライブラリの自動チューニング方式の確立を目指し、SR8000とRS/6000SPの1CPUにおいて、複数のアンロールの組について行列サイズを3000次元から4000次元まで1次元毎に評価した。

その結果、1次元毎の評価からは、低段アンロールでは見られなかった性能劣化が、高段アンロールにおいて同じ次元で見られたことが分かった。また、ピンポイントではループアンローリングの性能が入れ替わる場合があり、複数の点の平均をもってアンローリングの性能とすべきことを確認した。さらに、アンロール組のメモリ要求性能Byte/Flopと平均性能の関連を明らかにし、両計算機上で高性能な展開を得ることができた。

今後の課題はSR8000等のノード内・ノード間と組合せたハイブリッド環境における通信最適化方式の検討と性能評価、第二に、Cluster of SMP向けの自動チューニング方式の確立である。

参考文献

- [1] Ninf Project <http://ninf.apgrid.org/>
- [2] NetSolve project <http://www.cs.utk.edu/netsolve/>
- [3] ATLAS project <http://www.netlib.org/atlas/index.html>
- [4] project I-LIB <http://www.super-computing.org/~kuroda/nadia.html>
- [5] K. Naono, Y. Yamamoto, M. Igai, H. Hirayama: High performance implementation of tridiagonalization on the SR8000, HPC-ASIA2000, Beijing, pp.206-219, May. 2000.