

## [共同研究成果]

### 四面体有限要素法によるチャネル乱流解析

三浦 慎一郎 (東京都立産業技術高等専門学校 ものづくり工学科)

## 1 はじめに

複雑形状を有する流れ場の計算では、非構造格子（有限要素法では要素）を用いた計算が計算コストの点で有効な方法であると考えられる。しかしこのような複雑形状場の格子生成は、境界層を含む流れ場を伴うため莫大な格子点が必要となり、一般的には、解析対象領域を3D-CADによりモデル形状を作成し、汎用の格子生成アプリケーションを用いる必要がある。このとき有限要素法における非構造要素分割において、六面体の要素による完全な自動化は不可能な状況にあるが、四面体要素の場合にはほぼ全自動化が可能となりプリ処理の高効率化が期待できる。

線形四面体を用いた有限要素法の適用は、壁乱流のような壁面境界層を形成する流れ場に対して四面体要素のアスペクト比からなる補間精度の数値的不安定性が生じる問題が知られている<sup>[1],[2]</sup>。したがって、四面体要素を用いた流体計算は見られるものの、このような境界層を有する壁乱流計算についてはほとんど議論されていない。

そこで本研究ではこのような壁乱流計算における問題を回避するため、流速の補関数に気泡関数を有する四面体要素、圧力に関しての補関数に線形四面体要素を用いる。この要素は一般的にMINI要素と呼ばれている。

本研究では壁面方向の粘性項に陰解法を用いるため、計算付加が膨大になる。特に六面体要素に比べて、四面体要素では節点数に対して要素数が大きくなる。そのため、壁面付近における境界層などの解像度の確保をするためには多くの要素を必要とし、さらにアスペクト比の問題から、流れ方向への分割も十分行われなければならない。したがって、大規模な計算が必要となる。

前報<sup>[3]</sup>において、六面体要素を用いた構造格子での有限要素法による壁乱流計算を行い、並列化およびベクトル化による高効率な計算を行うことを報告した。有限要素法におけるベクトル化・並列化は必ずしも容易ではないと思われる。前報のようにベクトル演算率は十分に高くとも間接インデックスが多用される点で、計算能力はSX-7のピーク性能の20%程度に留まっている。また、前報では規則的な形状であり、規則的な節点番号を振ることで、再帰

呼び出しの無いループによりベクトル化が可能であった。

本計算では、非構造有限要素である四面体要素 (MINI 要素) を用いてベクトル並列計算機による計算の検討を行う。

## 2 基礎方程式及び数値計算法

### 2.1 気泡関数付き四面体要素

基礎方程式は、前報<sup>[3]</sup>同様に、非圧縮性粘性流体を仮定した Navier-Stokes 方程式と連続の方程式を用いる。壁面摩擦速度  $u_\tau$  及び平板間の半値で代表速度及び代表長さをとるものとして無次元化を行う。

計算のアルゴリズムに分離解法の一つである fractional step 法を用いる。空間に関する離散化は、一般的なガラーキョ有限要素法により定式化し、時間積分法として、対流項及び拡散項に 2 次精度 Adams-Bashforth 法を適用する。

運動方程式における対流項に関しては、発散型 (divergence form) を用いる。空間の補間関数に次に示す四面体線形要素  $N_\alpha$  および、四面体気泡関数付き補間関数  $N_\alpha^*$  を用いる。

$$N_\alpha = \frac{1}{6V} (a_\alpha + b_\alpha x_1 + c_\alpha x_2 + d_\alpha x_3) \quad (\alpha = 1 \sim 4) \quad (1)$$

$$N_\alpha^* = N_\alpha - \frac{1}{4} N_5^* \quad (\alpha = 1 \sim 4) \quad (2)$$

$$N_5^* = c N_1 N_2 N_3 N_4 \quad (c = 4^4) \quad (3)$$

これらの補間関数を用いて、流速に関する補間は気泡関数付き補間関数を用いて、圧力の補間に関しては、線形補間を用いる。図 1 に MINI 要素の概念図を示す。その結果、要素ごとに流速については 5 点、圧力に関しては 4 点で評価される。 $\alpha$  は四面体各頂点の数であり、 $a_\alpha, b_\alpha, c_\alpha, d_\alpha$  は四面体要素を構成する座標により決定される。

これらの補間関数を用いて質量行列、対流行列、拡散行列などの係数行列が作成される。各要素の係数行列は体積積分による解析的積分により求めるが、対流項のみ Gauss-Legendre 法に基づく数値積分公式 (45 点) により行う。各要素の係数行列はメモリ上に記憶され、時間ステップ中に係数行列の再計算は行わない。

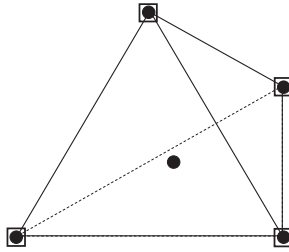


図 1: MINI Element (●:velocity nodal point,□:pressure nodal point)

## 2.2 並列化法及び行列の格納方法

有限要素法による行列の解法は，差分法と異なり，一般的には単純な  $n$  重対角行列とはならない．しかし疎行列であるので，連立方程式の解法に伴う行列ベクトル積の高速化が計算の効率化を図る上で重要となる．

有限要素法における連立方程式を解くための行列は疎行列となるが，節点番号の振り方によりそのバンド幅が大きく関係する．特に不規則要素では節点番号のばらつきに応じて非ゼロ成分を多く含み，バンド幅が大きくなる．

本計算においての節点番号は，四面体要素の頂点においてはある程度規則的に節点番号が振り分けているため散らばり具合が小さくされているが，流速を求める節点番号は通常四面体要素から汎用性を考慮し，気泡関数点を四面体節点番号から付け足す形にしているため，その値の振り方がバンド幅を大きくする．

この影響により四面体頂点で評価される圧力点に関しては，比較的バンド幅の小さい行列になるが，気泡関数点を含む流速値の行列はバンド幅の大きい行列となり，適切な疎行列の格納を考える必要がある．したがって圧力では Compressed-diagonal storage(CDS) 形式<sup>[4]</sup>を用い，流速に関する行列の解法は，バンド幅が大きいため CDS 形式の格納を行わず，Jugged-diagonal strage(JDS) 形式を基に，さらに 1 次元化への変換を行ったものを用いる．

図 2 に本計算で用いた行列の格納形式を示す．まず，CRS 形式により格納された行列を列の大きいものを行の上に来るように並べ替える．この行列を 1 次元化して行列のゼロ成分を取り除く．また 1 次元配列化を行う前の行方向での並列化が 1 次元配列化後も可能となる．

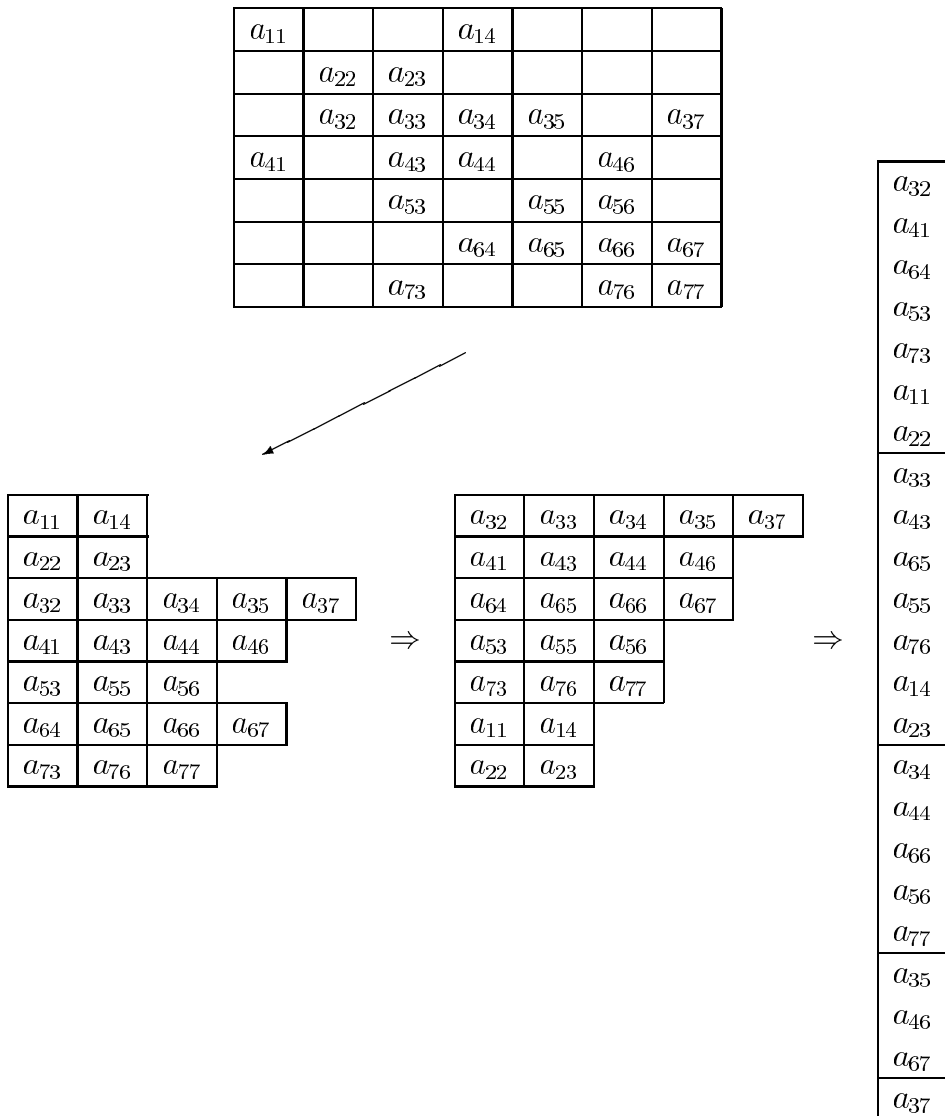


図2 行列の格納方法

本計算における行列の解法には、運動方程式に関してはBiCGStab法、圧力のポアソン方程式には共役残差法（CR法）を用いた、それぞれの前処理法は対角スケールリングを用いる。

```

!$omp parallel private(i,ista,iend,id_num)
!
      id_num = OMP_GET_THREAD_NUM()
!
      do j = 1, lr
!
          ista = iee(j)* id_num      /nb_threads + 1
          iend = iee(j)*(id_num + 1)/nb_threads
!
!$omp barrier
!cdir nodedp
          do i = ista, iend
              ap(i) = ap(i) + sa(i,j)*x(ia(i,j))
          enddo
!
      enddo
!$omp end parallel

```

図3 JDS形式の行列ベクトル積<sup>[4]</sup>

```

!$omp parallel private(i, k, k1, id_num)
!
      id_num = OMP_GET_THREAD_NUM()
!
      k1 = 0
      do j = 1, lr
          k1 = k1 + iee(j-1)
!$omp barrier
!cdir nodedp
          do i = is_col(j,id_num) , ie_col(j,id_num)
              k = k1 + i
              ap(i) = ap(i) + sa_col(k)*x(k)
          enddo
      enddo
!$omp end parallel

```

図4 並べ替え付き JDS形式の1次元化による行列ベクトル積

```

9486: V-----> do j = 1, lr
9487: |      S    k1 = k1 + iee(j-1)
9488: |          !$omp barrier
9489: |          !cdir nodep
9490: |V-----> do i = is_col(j,id_num), ie_col(j,id_num)
9491: ||          k = k1 + i
9492: ||          ap(i) = ap(i) + sa_col(k)*x(k)
9493: |V----- enddo
9494: V----- enddo
9595:          !$omp end parallel
    
```

図5 図4の並列ベクトル化診断プロファイル

図3のJDSによるプログラムは、ベクトル長を総節点数の長さで取れるため理想的なベクトル長であるが、間接インデクス（プログラム中の配列  $x$  に対するインデクス）により、ベクトル化演算率は高くとも演算速度が十分で無い場合がある。またこの方法では、非ゼロ成分も一部記憶し演算していることなどから、CRS法などに比べて記憶容量や余分な演算が増える欠点を持つ。

これらを改善するために図2で示された1次元化を図ったプログラムを図4に示す。

図4では、非ゼロ成分の記憶を行わず、さらに配列  $x$  のインデクスはスカラ変数の呼び出しにより連続的な参照を行うように改善している。

図5に図4で示したプログラムに対するベクトル・並列化診断プロファイルを示す。この結果から最内ループのみならず外側ループに対してもベクトル化が行われており、一ヶ所(9487行目)にスカラ演算が行われている所が見られるが、ここに配列を用いてスカラ演算を回避することが出来るが、最内ループで参照するため結果的に参照インデクスと変わらない結果となり有効な効果は得られなかった。

これらの改善により演算性能では、図3に示す2次元でのJDSでは、1Gflops程度（ベクトル化演算率99.2%）、図4に示す1次元化を図った方法では、2.2Gflops（ベクトル化演算率99.3%）となった。

### 3 計算条件及び計算結果

数値計算手法の有効性を検討するため、提案する手法を  $Re_\tau = 180$  を有するチャンネル乱流場の計算に適用する。計算領域は、主流方向を  $x_1$ 、垂直方向を  $x_2 (= y^+)$ 、スパン方向を  $x_3$  とし、その領域は  $4\pi\delta \times 2\delta \times (4/3)\pi\delta$  とする。

主流方向及びスパン方向には周期境界条件を課す。要素分割方法は垂直方向成分には  $\tanh$  関数よりなる不等長要素、他の方向には等分割とする。本計算ではメッシュ解像度の異なる2つの要素分割を行った。Mesh1として壁面方向のみ64分割し、主流、スパン方向は32分割したものと、Mesh2として、全方向64分割する。さらにそれぞれの六面体内に図6に示すように6個の四面体要素を構成する。ただし壁面方向およびスパン方向での対称性を考慮して、全体領域の四面体要素を配置する。また図6に示す四面体要素への分割の際に、規則的な節点番号の呼び出しを行うことがベクトル化・並列化を行う上で必要となる。

壁面における最小メッシュ幅は  $y^+ = 1$  で、このときの気泡関数点では  $y^+ = 0.25$  で与えられる。上下壁面の条件はすべり無し条件を与える。乱流統計量に関しては充分発達した乱流場から時間及び空間のアンサンブル平均から求める。それらのデータを Kim ら [5] の DNS データと比較する。

本計算は SX-7C(8CPU) による結果を示す。

図7は主流方向の流速成分を示している。Mesh2では、DNSの結果に比べると対数領域である壁面間の中心付近での流速が過小評価されていることが分かるが、粗いメッシュ(Mesh1)に比べると壁面付近の直線低層 ( $y^+ < 5$ ) およびバッファー層 ( $y^+ < 40$ ) 付近まで正確に捉えられていることが分かる。

図8は各方向の乱れ成分強度を示す。壁面方向でのメッシュ解像度は十分であり、Mesh1, Mesh2のいずれも良好な結果が得られている。主流方向への乱れ成分が過大評価されていることから、メッシュ解像度の改善の必要がある。

表1に本計算でのプロファイル情報を示す。SX-7Cの使用時における主なコンパイラオプションは、'-sx8 -Chopt -Popenmp' を用いている。時間ステップは200,000である。表1は入出力を含めての情報である。全体としてベクトル演算率は99%を超えており、良好なベクトル化が行われているが、間接インデックスの影響により全体での演算速度が1GFLOPS程度しか得られていないことが分かる。バンクコンフリクトの時間も計算時間との割合から比べると比較的大きいものになっているため、今後検討が必要になる。並列化は

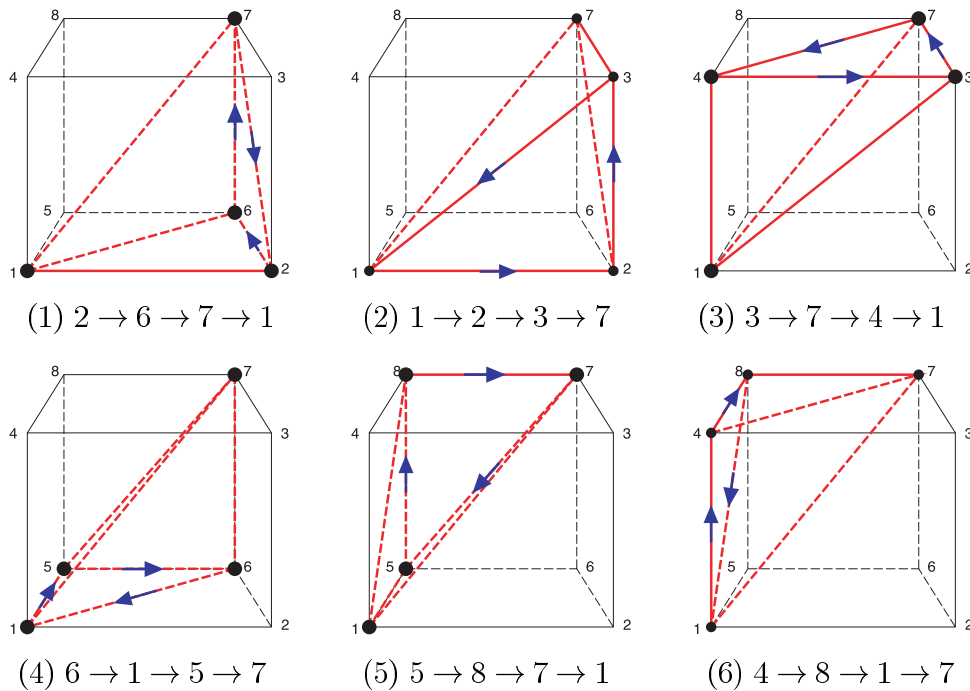


図6 六面体要素から四面体要素への分割方法

良好に行われていることが分かる。

## 4 まとめ

本研究では、壁面摩擦速度に基づく Reynolds 数 180 におけるチャンネル乱流計算を行った。本計算においては、乱流モデルは導入していない。数値計算法は、空間に関しては気泡関数付き補間関数による四面体 Galerkin 有限要素法を用いた。特に対流項に対して発散型により離散化を行い、質量行列には multi-pass アルゴリズムを用いた。時間積分法には壁面による影響を強く受けるため、壁面方向の物理粘性項には Crank-Nicolson 法による陰解法を適用した。その他は 2 次精度 Adams-Bashforth 法による陽解法、圧力 Poisson 方程式には後退 Euler 法を適用した。

ベクトル型並列スーパーコンピュータの利用を考慮した高速化及び並列化を行うため、大規模連立方程式の解法には、1 次元化された JDS 形式に基づく格納形式により行列ベクトル積の高速化・並列化を図った。解法には運



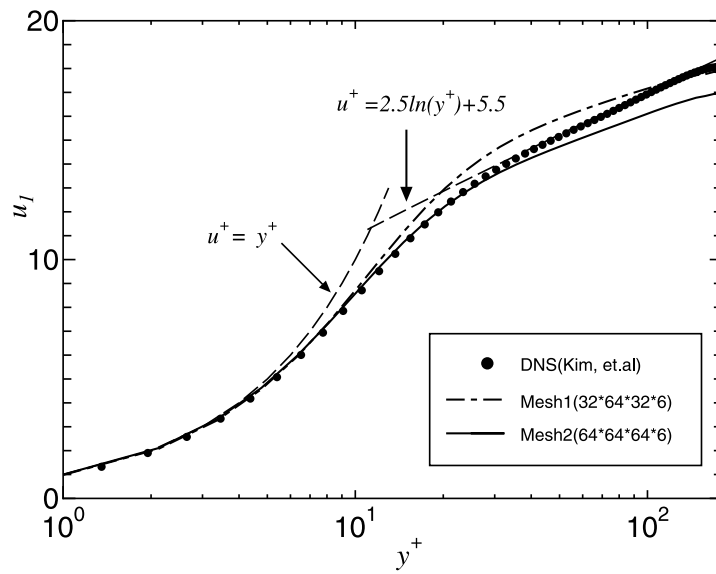


図7 主流方向流速成分

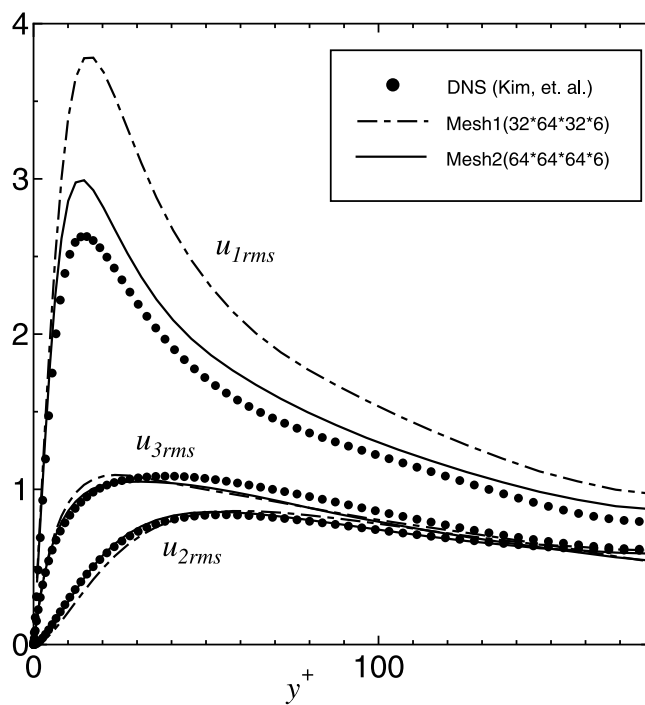


図8 各方向乱れ強度分布

動方程式に対しては BiCGstab 法, 圧力のポアソン方程式には CR 法を用いた。六面体の場合は運動方程式に対しても CR 法を用いた方が効率的であったが, 四面体要素を用いた場合では収束回数が多くなり計算時間が掛かったため, BiCGstab 法を用いた方が効率が良かった。

今後の課題として, 今回の計算では非構造要素に基づくものであるが, 完全に不規則な要素を用いていない。要素の呼び出し順序を特定して, ベクトル化を行っている。より複雑な形状に関しては CAD データなどから得られた形状を, メッシュジェネレータなどの汎用ソフトによって分割された要素は, 不規則な節点データであることが考えられる。そのようなデータに関しても, 対応できるようにする必要がある。

**謝辞:** 本研究で使用されたベクトル並列計算機, NEC SX-7C を用いた計算の高速化・並列化については, 東北大学情報シナジーセンター (現 東北大学サイバーサイエンスセンター) 各位にご指導を頂きました。ここに記して感謝の意を示します。

## 参考文献

- [1] 郭陽, 加藤千幸ら, ベンチマーク流れの有限要素法による LES 解析, 第 18 回数値流体力学シンポジウム, A-3, 2004.
- [2] 山井吉伸, 加藤千幸, 郭陽, 塚本裕一, 四面体要素による乱流解析, 生産研究, **59-1**, 2007, 77-82.
- [3] 三浦慎一郎, 有限要素法によるチャンネル乱流の LES 解析, SENAC, **40-2**, 2007, 15-29.
- [4] G. R. Brozolo, M. Vitaletti : Conjugate gradient subroutines for the IBM 3090 Vector Facility, *IBM J. DEVELOP.*, Vol.33, No. 2, pp.125-135, 1989.
- [5] Kim, J., Moin, P. and Moser, R., Turbulence Statistics in Fully Developed Channel Flow at Low Reynolds Number, *J. Fluid Mech.*, **177**, 1987, 133-166

表1 実行プロファイル情報(SX-7C, 8CPU), 200000 タイムステップ

***** Program Information *****	
Real Time (sec)	: 86941.262527
User Time (sec)	: 666916.372265
Sys Time (sec)	: 1466.212178
Vector Time (sec)	: 301225.735499
Inst. Count	: 32368125788402.
V. Inst. Count	: 11114316435195.
V. Element Count	: 2843898142634626.
FLOP Count	: 699003845143041.
MOPS	: 4296.118781
MFLOPS	: 1048.113188
MOPS (concurrent)	: 33243.842006
MFLOPS (concurrent)	: 8110.415705
VLEN	: 255.877018
V. Op. Ratio (%)	: 99.258196
Memory Size (MB)	: 14272.000000
Max Concurrent Proc.	: 8.
Conc. Time(>= 1)(sec):	86185.945399
Conc. Time(>= 2)(sec):	83286.487642
Conc. Time(>= 3)(sec):	83286.136436
Conc. Time(>= 4)(sec):	83286.051462
Conc. Time(>= 5)(sec):	83285.977736
Conc. Time(>= 6)(sec):	83283.771360
Conc. Time(>= 7)(sec):	83206.578507
Conc. Time(>= 8)(sec):	81095.876871
Event Busy Count	: 0.
Event Wait (sec)	: 0.000000
Lock Busy Count	: 22.
Lock Wait (sec)	: 0.043266
Barrier Busy Count	: 0.
Barrier Wait (sec)	: 0.000000
MIPS	: 48.534010
MIPS (concurrent)	: 375.561533
I-Cache (sec)	: 990.397836
O-Cache (sec)	: 15685.067431
Bank (sec)	: 14985.601575