



JP0150170

JAERI-Review

2000-019



次世代計算機概念検討
—平成11年度作業報告—

2000年12月

樋口 健二・長谷川 幸弘・平山 俊雄

日本原子力研究所
Japan Atomic Energy Research Institute

本レポートは、日本原子力研究所が不定期に公刊している研究報告書です。

入手の問合せは、日本原子力研究所研究情報部研究情報課（〒319-1195 茨城県那珂郡東海村）あて、
お申し越しください。なお、このほかに財団法人原子力弘済会資料センター（〒319-1195 茨城県那珂郡
東海村日本原子力研究所内）で複写による実費頒布をおこなっております。

This report is issued irregularly.

Inquiries about availability of the reports should be addressed to Research Information Division,
Department of Intellectual Resources, Japan Atomic Energy Research Institute, Tokaimura, Naka-
gun, Ibarakiken 319-1195, Japan.

次世代計算機概念検討
－平成 11 年度作業報告－

日本原子力研究所計算科学技術推進センター

樋口 健二・長谷川 幸弘・平山 俊雄

(2000 年 9 月 22 日受理)

計算科学技術推進センターでは、並列処理に関わる基盤技術開発の一環として、平成 11 年度に次世代計算機の概念検討に着手した。ここで行った代表的な量子計算コードを対象とした挙動解析結果について述べる。また、解析結果に対する考察、キャッシュミス緩和のための方策の検討結果についても触れる。さらに、概念検討を定量的に進めるために開発中の性能予測シミュレータについても述べる。

Examination of Concept of Next Generation Computer
Progress Report 1999

Kenji HIGUCHI, Yukihiro HASEGAWA, Toshio HIRAYAMA

Center for Promotion of Computational Science and Engineering
Japan Atomic Energy Research Institute
Nakameguro, Meguro-ku, Tokyo

(Received September 22, 2000)

The Center for Promotion of Computational Science and Engineering has conducted R&D works on the technology of parallel processing and has started the examination of the next generation computer in 1999. This report describes the behavior analyses of quantum calculation codes. It also describes the consideration for the analyses and examination results for the method to reduce cache misses. Furthermore, it describes a performance simulator that is being developed to quantitatively examine the concept of the next generation computer.

Keywords: Next Generation Computer, Vector Processing, Parallel Processing,
Numerical Simulation

目 次

1. はじめに	1
2. 熱伝導-対流遷移コード	4
2.1 コード概要	4
2.2 動的挙動解析結果と専用回路の適用可能性	5
2.3 キャッシュ・チューニング	9
3. 溶融塩粘性率計算コード	13
3.1 コード概要	13
3.2 動的挙動解析結果と専用回路の適用可能性	14
3.3 キャッシュ・チューニング	19
4. 三次元ジャイロ運動論粒子コード	22
4.1 コード概要	22
4.2 動的挙動解析結果と専用回路の適用可能性	22
4.3 キャッシュ・チューニング	27
5. 性能シミュレータ・プロトタイプの開発	33
5.1 MD0 コードにおける計測箇所	33
5.2 性能予測における仮定	35
5.3 性能予測に使用したモデル	35
5.4 評価結果	38
6. まとめ	43
6.1 量子計算コードの挙動解析	43
6.2 性能シミュレータ	43
6.3 今後の課題	44
参考文献	45
付録A MD0 の ANALYZER-P による解析結果	46
付録B MD1 の ANALYZER-P による解析結果	49
付録C GYRO3D の ANALYZER-P による解析結果	53
付録D キャッシュミス影響予測子 Cf と計測結果	58
付録E 性能シミュレータの使用方法	65

Contents

1. Introduction	1
2. Monte Carlo Simulation for Transition between Heat Conduction and Convection	4
2.1 Outline of the Code	4
2.2 Behavior Analysis and Possibility for Special Hardware	5
2.3 Tuning Technique to Reduce Cash Misses	9
3. Molecular Dynamics for Calculation of Shear Viscosity of Molten Salt	13
3.1 Outline of the Code	13
3.2 Behavior Analysis and Possibility for Special Hardware	14
3.3 Tuning Technique to Reduce Cash Misses	19
4. Gyrokinetic Particle Simulation	22
4.1 Outline of the Code	22
4.2 Behavior Analysis and Possibility for Special Hardware	22
4.3 Tuning Technique to Reduce Cash Misses	27
5. Development of the Prototype of Performance Simulator	33
5.1 Measurement of MD0 Code	33
5.2 Hypotheses in Performance Prediction	35
5.3 Simulation Model	35
5.4 Simulation Results	38
6. Summary	43
6.1 Behavior Analyses for Quantum Calculation Codes	43
6.2 Performance Simulator	43
6.3 Future Work	44
References	45
Appendix A Behavior Analysis of MD0 Code by ANALYZER-P	46
Appendix B Behavior Analysis of MD1 Code by ANALYZER-P	49
Appendix C Behavior Analysis of GYRO3D Code by ANALYZER-P	53
Appendix D Predictor for Affairs of Cash Misses	58
Appendix E User's Guide of Performance Simulator	65

1. はじめに

近年、マイクロ・プロセッサ(MPU)の性能向上は著しい。従来多大の演算量と主記憶容量を要するためスーパーコンピュータ上でのみ実行可能であった科学技術計算の多くが、ワークステーションやパーソナルコンピュータ上で処理可能となりつつある。そのため、応用ソフトウェアは、これらの計算機上で開発され、また、広く流通する傾向がますます強まっていく。すなわち、科学技術計算においても、汎用 MPU をベースとして、これに特殊機能を追加することで更なる高速化を追及することが今後のひとつの方向となる。次世代計算機についても、特殊演算回路を装備した汎用 MPU を要素計算機とする並列計算機を想定して概念検討を進めた。高速化する分野については、流体計算については地球シミュレータが、また、粒子計算のうちモンテカルロ・コードについては既に高速モンテカルロ装置が適合することから、粒子輸送モンテカルロ・コード以外の粒子計算（以下、本報では量子計算コードと呼ぶ）を対象に概念検討を進めた。ここでは、演算性能とデータ供給性能の二つの観点から、量子計算コードを高速化するためのハードウェアの概念検討及びその可能性について検討を行った。

演算性能を向上させる手法については、通常、二つの手段がある。一つは、ベクトル処理や並列処理等の既存のアーキテクチャの特性を引き出す方向でプログラミングするものである。いま一つは、ベクトル／並列演算機構が適用できない、あるいは単体性能を向上させたい場合であり、専用回路による高速化である。浅井らは、この観点から、モンテカルロ装置を開発し、良好な結果を得た[1]。浅井らは、モンテカルロ・コードにおいて多用され、かつ、ベクトル／並列演算が適用できない、あるいは適用しても高速化できない計算処理を高速処理するためのモンテカルロ・パイプラインと呼ばれる専用回路を開発し、これをベクトル並列計算機に搭載し、従来のベクトル計算機において 2 倍程度であった速度向上率を 10～20 倍程度に引き上げた。

次世代計算機についても、モンテカルロ装置同様、専用回路による高速化を検討中である。量子計算に関わる処理内容は、たかだか有限個の関数群の部分集合で大半を覆うことができるものと予想される。例えば、第一原理による分子動力学計算については高速フーリエ変換、光科学分野の計算については三角関数やローレンツ因子計算等の関数が多用され、しかも、主要な計算コストを占めている。そこで、量子計算コードで多用される、高速フーリエ変換やローレンツ因子計算等の関数を高速化するための専用演算回路を開発、並列計算機

に実装し、演算性能を改善する可能性を検討中である。平成 11 年度は、実用の量子計算コードにおいて多用される関数群を特定するための挙動解析を行った。

一方、データ供給性能の検討については、本来、キャッシュや TLB（アドレス変換バッファ）の容量や戦略等を考慮した、メモリアクセスのバンド幅に関する総合評価が必要であるが、今回は、演算回路へデータをロードする際性能低下の決定的要因になりがちなキャッシュ・ミスヒットの改善策を検討した。ここでは、実用の量子計算コードを対象に、キャッシュ・ミスヒット緩和のためのアルゴリズムを検討した。

キャッシュ・ミスヒットの緩和については、これまで多くの研究がなされてきた。それらの研究は、ハードウェア設計上の問題を取り扱ったものとデータの参照予測アルゴリズムを提案するものに分類することができる。前者については、例えば、細見ら[2]は、分散共有メモリ型並列計算機におけるネットワークトラフィックの削減及びメモリアクセスレイテンシの削減／隠蔽という観点から、Invalidate 型、Update 型、Competitive 型の 3 つのキャッシュコヒーレンスプロトコルの性能評価を行った。Rotenberg ら[3]は、命令長が大きくなり命令のキャッシュミスが多発するスーパースカラ・プロセッサのために、動的な命令列をトレースするトレース・キャッシュの効果を示した。また、キャッシュミスの改善とは異なるが、前述のモンテカルロ装置の開発において、浅井らは、粒子輸送モンテカルロ・コードで多用される間接番地参照データの主記憶からベクトル・レジスタへの転送能力を強化することでモンテカルロ・コードの高速処理に有効な手法を示した[1]。この種の研究は、ハードウェア構造に大きく依存する問題を取り扱っているため、計算機アーキテクチャが常に変遷する現状では議論が尽きることのない分野である。後者すなわち参照予測アルゴリズムについては、例えば、Joseph ら[4]は、主記憶等からキャッシュへデータを効率良くプリフェッチするための戦略としてマルコフ・プリフェッチャを提案した。このプリフェッチャは、プログラム中の各変数の参照関係にマルコフ性を当てはめ（各変数は決まったパターンで参照されているものと想定し）、その参照履歴から次のステップで使用される変数を予測するものである。Joseph らは、シミュレータを用いて、提案した一次のマルコフ・プリフェッチャに対して性能評価を行い、さまざまな商用のベンチマークにおいて命令とデータに関するメモリ操作によるストールを平均 46% 削減した。本研究では、ハードウェア構造にさほど依存しない、この種の研究分野を対象としている。例えば、計算機の主記憶及びキャッシュを、それぞれ、人間の長期

記憶及び短期記憶に当てはめ、人間の記憶モデルを参考にデータの参照頻度関数を構築することを検討中である。次段階では、これを用いてそれぞれの記憶領域上におけるデータの存在確率を予測すると共に、最適なデータ配置を行う手法を検討する予定である。Wickelgren は、忘却は時間と干渉によって起こり、記憶の痕跡は記憶の強さともろさという 2 つの動的性質によって説明づけられるとして、記憶の強さともろさを学習後の時間の関数として表現した[5]。この Wickelgren の記憶保存関数を用いて、藤井らは、大型計算機の運用システムを構築する際、現用ニーズに最適な記憶媒体の容量を決定した[6]。本研究においては、藤井らの手法を参考に、Wickelgren の記憶保存関数を利用したプリフェッチやキャッシュ・主記憶間のデータ転送の最適化等を現在検討中である。平成 11 年度は、実用の量子計算コードを対象に、キャッシュ・ミスヒットをアルゴリズム・レベルのチューニングによって緩和する手法の効果を検証した。

2 章では熱伝導-対流遷移コード MD0 について、3 章では溶融塩粘性率計算コード MD1 について、4 章では三次元ジャイロ運動論粒子コード GYRO3D について、それぞれ、コードの挙動解析結果及びキャッシュ・チューニング結果について述べる。また、専用回路の適用可能性についても触れる。5 章では、次世代計算機の性能予測を定量的に進めるために開発中の性能シミュレータのプロトタイプについて述べる。

2. 熱伝導-対流遷移コード

2.1 コード概要

熱伝導-対流遷移コード MD0[7]は、流れ場が熱伝導から対流状態へ移行する際の流体を構成する原子の運動を調べるために、決定論的手法であるMD(Molecular Dynamics)法[8]を用いて熱伝導-対流遷移過程をシミュレーションする。処理の中心は各粒子間に働く力の計算である。粒子 i の受ける力を F_i 、粒子の位置を r_i とすると、運動方程式は次式で表すことができる。

$$\frac{d^2 r_i}{dt^2} = \frac{F_i}{m}, i = 1, 2, \dots, N$$

原子間の距離が r のときのポテンシャルエネルギーを ϕ_{ij} とすると、粒子 j が粒子 i に及ぼす力 F_{ij} は次式で表すことができる。

$$F_{ij} = -\nabla_i \Phi_{LJ}(r_{ij})$$

ここで、ポテンシャルエネルギーは、レナード・ジョーンズ・ポテンシャルであるから、 F_{ij} と ϕ_{ij} は次式で表すことができる。

$$F_{ij} = 24\epsilon \left\{ 2 \left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right\} \frac{1}{r_{ij}}$$

$$\Phi_{LJ}(r) = 4\epsilon \left\{ \left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right\}$$

ここで、 $r_{ij} = r_i - r_j$ ， $\sigma = 1$ ， $\epsilon = 1/48$ である。

上記の式を用いて、各粒子に対して他のすべての粒子との間に働く力を計算する。この計算に消費する処理時間は粒子数のほぼ 2 乗に比例して増加するため、処理時間を短縮する工夫が必要である。MD0 コードでは、距離による粒子カットオフと粒子登録法を使用している。

カットオフは、ある距離以上離れた粒子間の相互作用はゼロと仮定する。すなわち、カットオフ距離以上離れた粒子同士の相互作用を計算対象外とする。

粒子登録法は、MD 法において、粒子が 1 ステップで移動する距離は微小であることを利用する。Figure 2.1.1 に示すように、カットオフ距離($rcut$)よりも長い距離($rint$)を定義し、この内側にいる粒子の粒子名を、相互作用しそうな粒子として相互作用粒子リストに格納する。たとえカットオフ距離内への粒子の出入りがあっても、 $rint$ 内の粒子を把握しておき、ある程度の間はその粒子の情報をそのまま使い続けることができると仮定する。このリストの更新は

適宜行う。ここで、 r_{int} の値や何ステップ毎に情報を更新するかは、試行錯誤的に決めている。今回はリストの作成は計算の最初の 1 回だけとしている。

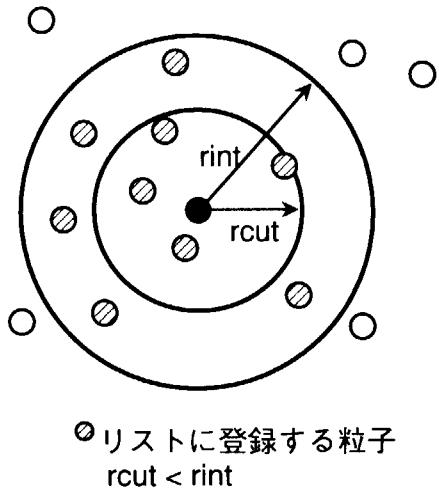


Figure 2.1.1 粒子登録法

2.2 動的挙動解析結果と専用回路の適用可能性

(1) 動的挙動解析結果

Table 2.2.1 に SX-4 における性能評価ツール Analyzer-P[9]を用いた挙動解析の結果を示す。付録 A に解析リストを示す。全処理時間が 734.805 秒であり、そのうちの 94.7%がサブルーチン TABLE と FORCE に集中している。TABLE は、各粒子ごとに自分と相互作用する相手粒子の名前を相互作用粒子リストに保存する。FORCE は TABLE で生成したリストを使って各粒子間に働く力を計算する。TABLE は、Fig. 2.2.1 に示す DO ループ 100 に全処理時間の 70.8%が集中している。FORCE は、Fig. 2.2.2 に示す DO ループ 271 に全処理時間の 21.9%が集中している。すなわち、この 2 つの DO ループが全処理時間の 92.7%を占め、MD0 の計算コアと考えられる。

キャッシュミスヒット時間は、インストラクションで 18.933 秒、オペランドで 78.046 秒である。オペランド・キャッシュミスはやや多く、全処理時間の約 10%を占めている。インストラクション・キャッシュミスは全処理時間の約 2%なので、許容範囲とみる。処理時間と同様、オペランド・キャッシュミスは TABLE の DO ループ 100 と FORCE の DO ループ 271 に集中している。

TABLE の DO ループ 100 は、粒子 i と粒子 j の距離を計算し、これが r_{init2} 以下になる粒子 j を相互作用粒子リスト配列 $itabs$ の粒子 i の項目に追加する。このループは先にも述べたように全体の約 70%もの処理時間を占めているが、

演算は 2 つの粒子間の距離 rx の計算のみである。ただし、(粒子数×粒子数) ケースの計算が発生して、演算回数が多い（今回のデータでは 3 199 960 000 回）ために処理時間がかかっている。

キャッシュは、局所性原理、すなわち、1) 一度参照された場所は近いうちに再び参照される可能性が高いという時間的局所性、2) 次に参照する場所は次の番地を含めて直前に参照した場所の近くである可能性が高いという空間的局所性、を利用して高い性能を得ることができる。DO ループ 100 の X(J) と Y(J) のサイズは、Table 2.2.2 に示したようにそれぞれ 640KB あり、64KB のオペランド。キャッシュに対してデータサイズが大きく、ずっとキャッシュに載っていることはない。キャッシュにフェッチされたデータは、再び参照される前に他のデータによって置換されて、実際参照したいときにはキャッシュに無くなっている可能性が高く、その場合は時間的局所性を使えない。

FORCE の DO ループ 271 は、相互作用粒子リストを使って粒子 i と粒子 j の距離を計算し、カットオフを行い、カットオフで生き残った粒子との間の力を計算している。このループのオペランド・キャッシュミスも TABLE の DO ループ 100 と同じ理由で発生していると考えられる。Table 2.2.3 に DO ループ 271 で使う配列のサイズを示す。ここでも 64KB のオペランド・キャッシュに対してデータサイズが大きいため、キャッシュミスが発生している。

Table 2.2.1 処理時間およびキャッシュミスヒット時間

Subroutine	CPU TIME		Inst. Cache Miss	Oper. Cache Miss		
TOTAL	734.805 sec.	100.0%	18.933 sec.	2.6 %	78.046 sec.	10.6 %
TABLE	521.822 sec.	71.0 %	0.758 sec.	0.0 %	61.575 sec.	8.0 %
FORCE	174.049 sec.	23.7%	7.477 sec.	1.0 %	12.815 sec.	2.0 %
CHKBND	12.018 sec.	1.6%	7.962 sec.	1.0 %	0.292 sec.	0.0 %
PCROSS	11.074 sec.	1.5%	1.078 sec.	0.0 %	0.931 sec.	0.0 %
PROPCEL	6.629 sec.	0.9%	0.362 sec.	0.0 %	0.655 sec.	0.0 %

```

do 101 i=1,ipmax

xi = x(i)

yi = y(i)

icol$ = 0

do 100 j=i+1,n

rx = (x(j)-xi)**2+(y(j)-yi)**2

if (rx.lt.rint2) then

    icol$ = icol$ +1

    itab(icol$,i ) = j

end if

100 continue

icol(i)=icol$

if(imax.lt.icol(i)) imax=icol(i)

call gettod(t2$)

101 continue

```

Figure 2.2.1 サブルーチン TABLE のコストの高い DO ループ

Table 2.2.2 サブルーチン TABLE の DO ループ 100 の配列サイズ

配列名	型	サイズ	
X(*)	Real*8	8 バイト × 80000 = 640KB	粒子の X 座標
Y(*)	Real*8	8 バイト × 80000 = 640KB	粒子の Y 座標
ITABS(*, *)	Int	4 バイト × 40 × 80000 = 12.8MB	相互作用粒子リスト

注) 粒子数 n=80 000 の場合の配列サイズを示す.

```

do 270 i=1,ipmax

xi = x(i)

yi = y(i)

fxi$=0.d0

fyi$=0.d0

do 271 k=1,icol(i)

j = itab(k,i)

xx = xi - x(j)

yy = yi - y(j)

rd = xx * xx + yy * yy

if ( rd .gt. rcut2 ) goto 271

rdr = 1./rd

rd3 = rdr**3

rd4 = rdr**4

rd = (rd3-0.5)*rd4

fxx = xx * rdrd4

fxi$ = fxi$ + fxx

fx(j) = fx(j) - fxx

fyy = yy * rd

fyi$ = fyi$ + fyy

fy(j) = fy(j) - fyy

271 continue

fx(i) = fx(i) + fxi$

fy(i) = fy(i) + fyi$

270 continue

```

Figure 2.2.2 サブルーチン FORCE のコストの高い DO ループ

Table 2.2.3 サブルーチン FORCE の DO ループ 271 の配列サイズ

配列名	型	サイズ	
X(*)	Real*8	8 バイト × 80000 = 640KB	粒子の X 座標
Y(*)	Real*8	8 バイト × 80000 = 640KB	粒子の Y 座標
ITABS(*,*)	Int	4 バイト × 40 × 80000 = 12.8MB	相互作用粒子リスト
ICOL(*)	Int	4 バイト × 80000 = 320KB	リストの登録数
FX(*)	Real*8	8 バイト × 80000 = 640KB	粒子間の力
FY(*)	Real*8	8 バイト × 80000 = 640KB	粒子間の力

注) 粒子数 n=80 000 の場合の配列サイズを示す.

(2) 専用回路の適用可能性

Table 2.2.4 に MD0 の主要ループ (サブルーチン Table 中の Do ループ 100 及びサブルーチン Force 中の Do ループ 271) のベクトル処理性能を示す. ここで, Do ループ 100 の平均ベクトル長は 40,000, Do ループ 271 の平均ベクトル長は 67 である. Do ループ 271 の速度向上率が Do ループ 100 と比較して低い理由は, ループ長が短いこと, 間接番地を使用していること, 及び条件分岐文が原因である. Do ループ 100 については, ベクトル処理の結果からもわかるように専用回路化によりベクトル処理同様の高速化が見込める. Do ループ 271 については, ループ長がやや短いものの, 汎用的なループ処理を行うベクトル処理と比べて立ち上がり時間の短い専用回路の場合, 十分な性能が期待できる.

Table 2.2.4 MD0 の主要ループのベクトル処理性能

	スカラ処理時間(S)	ベクトル処理時間(V)	速度向上率(S/V)
Loop 100 in subroutine Table	520.432 sec.	22.043 sec.	23.610
Loop 271 in subroutine Force	161.166 sec.	46.870 sec.	3.439

2.3 キャッシュ・チューニング

2.2 で行ったオペランド・キャッシュミスの考察にもとづいて, キャッシュ・チューニングを行った.

(1) 概要

将来必要になるデータに対して意図的に早めに参照を行うことで, そのデータが実際に必要となった時に演算が待たされないようにプログラミングする (アルゴリズム・プリフェッч[10]). すなわち, キャッシュへのフェッチを先行して発行してキャッシュミスによる遅延サイクルを隠蔽する. 演算部とプリフェッチ部に依存関係がなければ, 2 つの処理は並列に実行される. 今回は実

験的に 16 番地先のデータをプリフェッチした効果を見た。

(2) チューニング結果

キャッシング・チューニングによって、MD0 コードの処理時間とキャッシング・ミスヒット時間は Table 2.3.1 に示すように改善された。

Table 2.3.1 DO ループ 100 の処理時間とキャッシング・ミスヒットの変化

	修正前	修正後
処理時間	521.822 sec.	502.821 sec.
キャッシング・ミスヒット	61.084 sec.	51.498 sec.

Figure 2.3.1 にサブルーチン TABLE の DO ループ 100 のキャッシング・チューニング前のソース、Fig. 2.3.2 にキャッシング・チューニング後のソースを示す。

```
do 101 i=1,ipmax
      xi = x(i)
      yi = y(i)
      icol$ = 0
      do 100 j=i+1,n
          rx = (x(j)-xi)**2+(y(j)-yi)**2
          if (rx.lt.rint2) then
              icol$ = icol$ +1
              itab(icol$,i ) = j
          end if
100    continue
      icol(i)=icol$
      if(imax.lt.icol(i)) imax=icol(i)
      call gettod(t2$)
      continue
```

Figure 2.3.1 サブルーチン TABLE の DO ループ 100 (修正前)

```
do 101 i=1,ipmax
      xi = x(i)
      yi = y(i)
      icol$ = 0
      do 100 j=i+1,n
          xjxi=x(j)-xi
          yjyi=y(j)-yi
          dumxjxi=x(j+16)-xi
          dumyjyi=y(j+16)-yi
          rx = xjxi*xjxi+yjyi*yjyi
          if (rx.lt.rint2) then
              icol$ = icol$ +1
              itab(icol$,i ) = j
          end if
100    continue
      icol(i)=icol$
      if(imax.lt.icol(i)) imax=icol(i)
      call gettod(t2$)
101    continue
```

Figure 2.3.2 サブルーチン TABLE の DO ループ 100 (修正後)

3. 溶融塩粘性率計算コード

3.1 コード概要

溶融塩粘性率計算コード MD1[11]は、分子動力学法(MD)により溶融塩の粘性率を計算する。MD ステップを重ねて応用テンソルの非対角成分の相関関数を求め、久保の公式に基づく相関関数の時間積分から粘性率を得る。処理の中心は各粒子間に働く力とそれを求めるのに必要なポテンシャルエネルギーの計算である。

前章で述べた MD0 コードのレナード・ジョーンズ・ポテンシャルのような短距離オーダーの場合は、カットオフを導入して処理時間を短縮することができた。一方、MD1 では、長距離オーダーのポテンシャルであるクーロンポテンシャルを使うため、遠くの粒子との相互作用も考慮する必要がありカットオフは使えない。そのため、シミュレーション領域として一辺の長さが L の立方体のセルを考え、周期境界条件を用いる。Figure 3.1.1 に二次元の概念図を示す。周りのセルは基本セルを複写して作成した仮想セルである。ある粒子が境界を通ってシミュレーション領域から流出する場合、反対側の境界面を通してそのまま流入することを意味する。

基本セルは N 個の荷電粒子から構成されるとし、任意の粒子 i の電荷を q_i 、位置ベクトルを r_i で表す。クーロンポテンシャルによる基本セルの全エネルギー E は次のように表すことができる。

$$E = \frac{1}{2} \sum_n \left(\sum_{i=1}^N \sum_{j=1}^N \frac{q_i q_j}{|r_{ji} + Ln|} \right)$$

ここで、 $r_{ji} = r_j - r_i$, $n = (n_x, n_y, n_z)$, $n_x = 0, \pm 1, \pm 2, \pm 3, \dots$ である。 $N = (0, 0, 0)$ は基本セル、他の場合は周期境界条件による仮想セルを意味する。

処理を高速化するために、エワルドの方法(Ewald Method)を導入して n による収束を改善する。上記式はエワルドの方法を用いると次のように表すことができる。

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \left\{ \sum_n q_i q_j \frac{\operatorname{erfc}(\kappa |r_{ji} + Ln|)}{|r_{ji} + Ln|} + \frac{1}{\pi L^3} \sum_{k(\neq 0)} q_i q_j \frac{4\pi^2}{k^2} \exp(-k^2/4\kappa^2) \cos(k \cdot r_{ji}) \right\}$$

$$-\frac{\kappa}{\pi^{1/2}} \sum_{i=1}^N q_i^2$$

ここで、

$$\operatorname{erfc}(x) = 1 - \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt, k = 2\pi n/L, \kappa = 5/L$$

以上のように求めたクーロンポテンシャル・エネルギーを用いて粒子間に働く力や応用テンソルを計算する。粒子 i の質量を m_i 、粒子 i の X 軸および Y 軸方向の速度を V_{ix} と V_{iy} 、粒子 i の X 軸座標を r_{ix} 、粒子 i に働く力の y 成分を F_{iy} とすると、応用テンソルの非対角成分 J_{xy} は次式で表すことができる。

$$J_{xy} = \sum_{i=1}^N m_i V_{ix} V_{iy} + \sum_{i=1}^N r_{ix} \cdot F_{iy}$$

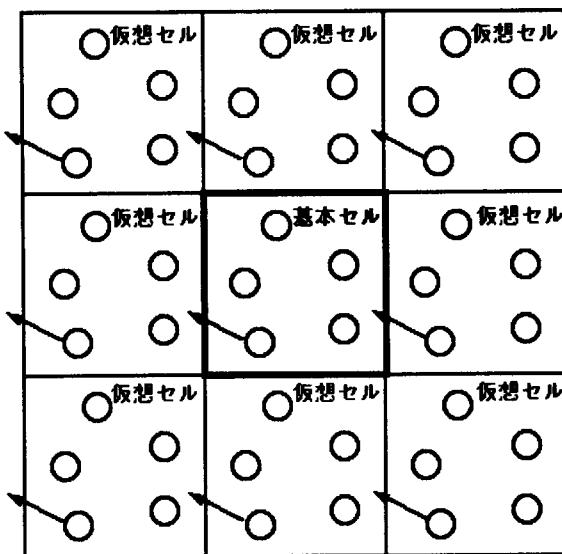


Figure 3.1.1 周期境界条件

3.2 動的挙動解析結果と専用回路の適用可能性

(1) 動的挙動解析結果

Table 3.2.1 に SX-4 における Analyzer-P を用いた挙動解析の結果を示す。付録 B に解析リストを示す。全処理時間が 5259.120 秒であり、そのうちの 97.0% が DO ループ 7000, 96, 199, 97 に集中している。Figure 3.2.1 に示す DO ループ 7000 は、粒子間の力 (TTF) と応用テンソル (TFI) を計算する。Figure

3.2.2 に示す DO ループ 98 が 3.1 で述べたエワルドの方法のループであり、この内側に DO ループの 96, 199, 97 がある。この中でも約 76% の処理時間を作る DO ループ 7000 が MD1 の計算コアである。DO ループ 7000 では、前もって計算しておいたボテンシャルエネルギーのテーブル(PE)や粒子間力のテーブル(RF)を参照することで処理時間の短縮を図っている。

キャッシュミスヒット時間は、インストラクションで 6.02 秒、オペランドで 391.557 秒である。インストラクション・キャッシュミスは全処理時間の約 0.1% なので許容範囲とみる。オペランド・キャッシュミスは全処理時間の約 7.4% である。そのうち DO ループ 7000 で 4%, DO ループ 96 で 3% であり、この 2 つのループでほとんど発生している。

以下、オペランド・キャッシュミスの原因について考える。まず、Table 3.2.2 及び Table 3.2.3 に示すようにデータ配列のサイズが大きいために、時間的局所性を使えていない。キャッシュにフェッチされたデータは、再び参照される前に他のデータによって置換されて、実際参照したいときにはキャッシュに無くなっている可能性が高い。また、配列アクセスのストライドが適当でないため、空間的局所性を使えていない。多次元配列の場合、アクセスのストライドが小さくなるように参照しないと、大きな性能低下を招く。MD1 はベクトル計算機用にチューニングがされているため、スカラー計算機にとっては逆に性能低下を招く結果となっている。MD1 をベクトル実行すると、ベクトル化率は 99.74% で処理時間は 288.251 秒となり、約 18 倍の性能向上がみられる。

Table 3.2.1 処理時間およびキャッシュミスヒット時間

DO Loop	CPU TIME		Inst. Cache Miss		Oper. Cache Miss	
TOTAL	5259.120 sec.	100.0%	6.026 sec.	0.1%	391.557 sec.	7.4 %
7000	4004.455 sec.	76.1%	3.086 sec.	0.0%	198.186 sec.	4.0%
96	556.573 sec.	10.6%	0.313 sec.	0.0%	158.299 sec.	3.0%
199	274.267 sec.	5.2%	0.408 sec.	0.0%	4.587 sec.	0.0%
97	270.815 sec.	5.1%	0.385 sec.	0.0%	10.049 sec.	0.0%

```

DO 7000 J = 1, NT
JON = KIM(J)
DO 7000 I = 1, NT
ION = KIMwork(I)
ZAA1 = Qwork(I,1) - Q(J,1)
ZAA2 = Qwork(I,2) - Q(J,2)
ZAA3 = Qwork(I,3) - Q(J,3)
IF( ZAA1 .GT. AR ) ZAA1 = ZAA1 - AL
      .
      .
      .
IF( ZAA3 .LT. AS ) ZAA3 = ZAA3 + AL
MF = IPR(ION,JON)
RR2 = ZAA1*ZAA1 + ZAA2*ZAA2 + ZAA3*ZAA3
IF( (RR2.LE.AR2) .AND. (I.NE.J) ) THEN
      R      = SQRT(RR2)
      N      = 100.0*R
      NH     = N/2+1
      EN2    = EN2 + PE(N, MF)
      RFA    = R - 0.01*REAL(N)
      ZEDQF  = 1.0E-04*(RF(N, MF)
      &           + 100.0*(RF(N+1, MF)-RF(N, MF))*RFA)
      ELSE
      ZEDQF = 0.0
      ENDIF
      TTF(J,1) = TTF(J,1) - ZEDQF*ZAA1
      TTF(J,2) = TTF(J,2) - ZEDQF*ZAA2
      TTF(J,3) = TTF(J,3) - ZEDQF*ZAA3
      TFI(J)   = TFI(J)   + ZEDQF*ZAA1*ZAA2
7000      CONTINUE

```

Figure 3.2.1 DO ループ 7000

```

DO 98 L=1,LEWA

VM1 = PDL*VN(L,1)

VM2 = PDL*VN(L,2)

VM3 = PDL*VN(L,3)

DO 97 I = 1, NL

      RV = VM1*Q(I,1)+VM2*Q(I,2)+VM3*Q(I,3)

      SI(I) = SIN(RV)*CHRG(1)

      CO(I) = COS(RV)*CHRG(1)

      XSI(I) = SIN(RV/AL)*CHRG(1)

      XCO(I) = COS(RV/AL)*CHRG(1)

CONTINUE

DO 199 I = NL+1,NT

      RV = VM1*Q(I,1)+VM2*Q(I,2)+VM3*Q(I,3)

      SI(I) = -SIN(RV)*CHRG(2)

      CO(I) = -COS(RV)*CHRG(2)

      XSI(I) = -SIN(RV/AL)*CHRG(2)

      XCO(I) = -COS(RV/AL)*CHRG(2)

199 CONTINUE

      DO 96 I = 1, NT

      ION = KIM(I)

      FACT = BM(NW(L),ION)*(TC0*SI(I)-TSI*CO(I))

      TTF(I,1) = TTF(I,1) + VN(L,1)*FACT

      TTF(I,2) = TTF(I,2) + VN(L,2)*FACT

      TTF(I,3) = TTF(I,3) + VN(L,3)*FACT

      TFI(I) = 0.0

96 CONTINUE

      98 CONTINUE

```

Figure 3.2.2 DO ループ 97, 199, 96

Table 3.2.2 DO ループ 96 の配列サイズ

配列名	型	サイズ	内 容
TTF(*,1-3)	Real*4	4バイト×4096×3 = 48KB	粒子間力
KIM(*)	Real*4	4バイト×4096 = 16KB	イオンのインデックス (1-CATION, 2-ANION)
VN(*,1-3)	Real*4	4バイト×800×3 = 9.6KB	セルの位置
SI(*)	Real*4	4バイト×4096 = 16KB	$\sin(k r_i)$
CO(*)	Real*4	4バイト×4096 = 16KB	$\cos(k r_i)$
BM(*,1-2)	Real*4	4バイト×80×2 = 0.6KB	Born-Mayer ポテンシャル項

注) 粒子数 n=4096 の場合の配列サイズを示す。

Table 3.2.3 DO ループ 7000 の配列サイズ

配列名	型	サイズ	内 容
Q(*,1-3)	Real*4	4バイト×4096×3 = 48KB	粒子の位置
Qwork(*,1-3)	Real*4	4バイト×4096 = 16KB	ワーク領域
IPR(*,*)	Real*4	4バイト×2×2 = 16B	Index Number of Ionic pair
PE(*,*)	Real*4	4バイト×3000×3 = 36KB	ポテンシャルエネルギーのテーブル
RF(*,*)	Real*4	4バイト×3000×3 = 36KB	原子間力のテーブル
TFI(*)		4バイト×4096 = 16KB	ストレス・テンソルのポテンシャル項
TTF(*,1-3)	Real*4	4バイト×4096×3 = 48KB	粒子間力

注) 粒子数 n=4096 の場合の配列サイズを示す。

(2) 専用回路の適用可能性

Table 3.2.4 に MD1 コードの主要ループ (Do ループ 7000, 96, 199 及び 97) のベクトル処理性能を示す。ここで、Do ループ 7000 及び Do ループ 96 の平均ベクトル長は 4,096, Do ループ 199 及び Do ループ 97 の平均ベクトル長は 2,048 である。Do ループ 96 のベクトル処理による速度向上率が高いのは、スカラー処理ではオペランド・キャッシュ・ミスヒットの発生原因となる配列が、大容量のベクトルレジスタにはうまくのるためである。Table 3.2.4 に示したすべての主要ループについて、専用回路化によりベクトル処理同様の高速化が見込める。

Table 3.2.4 MD1 コードの主要ループに対するベクトル処理性能

	スカラ処理時間(S)	ベクトル処理時間(V)	速度向上率 (S/V)
Loop 7000	4004.455 sec.	239.119 sec.	16.747
Loop 96	556.573 sec.	6.188 sec.	89.944
Loop 199	274.267 sec.	19.392 sec.	14.143
Loop 97	270.815 sec.	18.114 sec.	14.951

3.3 キャッシュ・チューニング

3.2 で行ったオペランド・キャッシュミスの考察にもとづいて、キャッシュ・チューニングを行った。

(1) 概要

本コードについては、主要ループで定義／参照される二次元配列の行と列を入れ替えてメモリ参照パターンを最適化することを試みた。

Figure 3.3.1 に二次元配列 $TTF(i,j)$ のメモリ上の並び、及び、行と列を入れ替えた場合のストライド（ある配列を一定間隔 S で定義／参照する時 S をストライドと呼ぶ）の変化を示す。

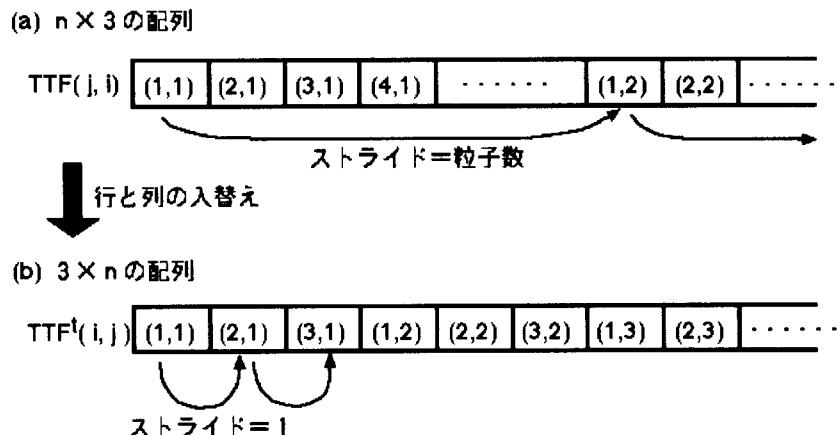


Figure 3.3.1 行と列の入替え

Figure 3.3.1 (a) のように、 n (粒子数) $\times 3$ の配列の場合、 $TTF(1,1) \rightarrow TTF(1,2) \rightarrow TTF(1,3)$ の順に参照すると、ストライド n (今回の入力データでは $n = 7000$) の不連続なメモリ参照となる。キャッシュへのデータのフェッチはライン単位に行われるが、ストライドがキャッシュのラインサイズ以上に大きい場合、次に参照されるデータは同一ライン上に載っていないので、次データを含むラインのフェッチが引き続き発生する (キャッシュ・ミスが起きる)。このように、ストライドが大きな場合、キャッシュミスヒットが頻繁に発生して性能低下を招く。これを改善するために配列の行と列を入れ替えて定義／参照するようコードを変更した。

すなわち、 $n \times 3$ の配列の行と列を入れ替えて $3 \times n$ の配列に変換すると、前述の TTF の参照は $TTF(1,1) \rightarrow TTF(2,1) \rightarrow TTF(3,1)$ の順になり、Fig. 3.3.1 (b)

に示すように、ストライドは 1 になる。この場合は連続なメモリ参照となり、 $TTF(1,j)$, $TTF(2,j)$, $TTF(3,j)$ はキャッシュの同一ラインにフェッチされるので、キャッシュを有効に利用できるようになった。

また、内側ループ内の同一ロード命令を外側に移動する（二重ループの外側ループで TTF, TFI を一時変数に置く）ことによって、内側ループでの TTF, TFI のロード／ストア命令を減らすことができた。

(2) チューニング結果

キャッシュチューニングによって、MD1 の処理時間とキャッシュ・ミスヒット時間は Table 3.3.1 のように改善された。

Table 3.3.1 DO ループ 7000 の処理時間とキャッシュ・ミスヒット時間の変化

	修正前	修正後
処理時間	4004.455 sec.	3928.581 sec.
キャッシュ・ミスヒット	198.186 sec.	118.640 sec.

Figure 3.3.2 に DO ループ 7000 のキャッシュ・チューニング前のソース、Fig. 3.3.3 にキャッシュ・チューニング後のソースを示す。

```

DO 7000 J = 1, NT
JON = KIM(J)
DO 7000 I = 1, NT
ION = KIMwork(I)
ZAA1 = Qwork(I,1) - Q(J,1)
.
.
.
TTF(J,1) = TTF(J,1) - ZEDQF*ZAA1
TTF(J,2) = TTF(J,2) - ZEDQF*ZAA2
TTF(J,3) = TTF(J,3) - ZEDQF*ZAA3
TFI(J)   = TFI(J)   + ZEDQF*ZAA1*ZAA2
7000 CONTINUE

```

Figure 3.3.2 DO ループ 7000 (修正前)

```
DO 7001 J = 1, NT
  JON = KIM(J)
  TTF1J = TTF(1,J)
  TTF2J = TTF(2,J)
  TTF3J = TTF(3,J)
  TFIJ=TFI(J)
DO 7000 I = 1, NT
  ION = KIMwork(I)
  ZAA1 = Qwork(I,1) - Q(J,1)
  .
  .
  .
  TTF1J = TTF1J - ZEDQF*ZAA1
  TTF2J = TTF2J - ZEDQF*ZAA2
  TTF3J = TTF3J - ZEDQF*ZAA3
  TFIJ = TFIJ + ZEDQF*ZAA1*ZAA2
7000 CONTINUE
  TTF(1,J)=TTF1J
  TTF(2,J)=TTF2J
  TTF(3,J)=TTF3J
  TFI(J)=TFIJ
7001 CONTINUE
```

Figure 3.3.3 DO ループ 7000 (修正後)

4. 三次元ジャイロ運動論粒子コード

4.1 コード概要

三次元ジャイロ運動論粒子コード GYRO3D[12]は、粒子の運動をジャイロ軌道（円軌道）で平均化した方程式系を用いるジャイロ運動論モデル(Gyrokinetic Model)を用いて、トカマクプラズマ中の荷電粒子の運動論的(巨視的)な振舞いを調べる。GYRO3Dは、粒子電荷/電流からメッシュ点上の電荷/電流を計算する Charge Deposit 部、メッシュ点上の電荷/電流から電場/磁場のポテンシャルを計算し、さらに電場/磁場の計算を行う Field Solver 部、求めた電場/磁場の値を使用して各粒子の新しい位置・速度を計算する Particle Pushing 部の3つの処理で構成されている。Figure 4.1.1に、Charge Deposit 部と Particle Pushing 部の計算に関する二次元の概念図を示す。

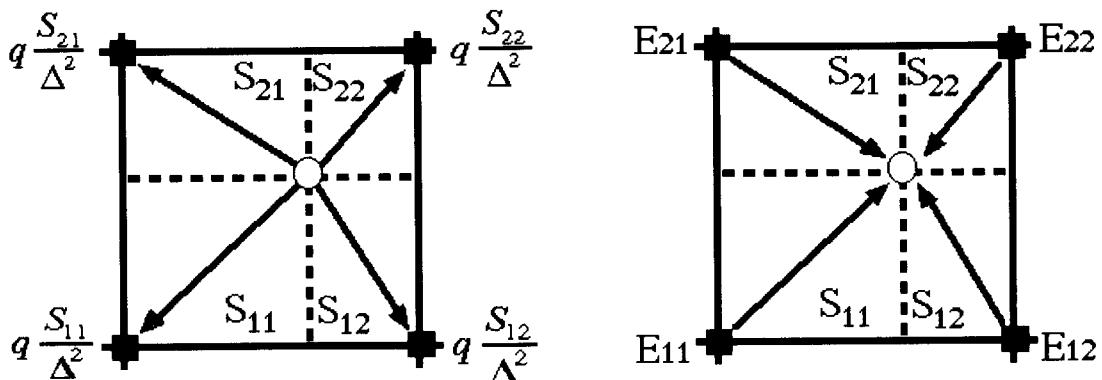


Figure 4.1.1 GYRO3D の各メッシュにおける計算概念

4.2 動的挙動解析結果と専用回路の適用可能性

(1) 動的挙動解析結果

Table 4.2.1 に SX-4 における Analyzer-P を用いた挙動解析の結果を示す。付録 C に解析リストを示す。全処理時間が 133.112 秒であり、そのうちの 71.2% がサブルーチン SRC, FFT2B, PSH2, USINM, PSH1 に集中している。SRC は Charge Deposit 部、FFT2B と USINM は Field Solver 部、PSH1 と PSH2 は Particle Pushing 部のサブルーチンである。ここで、粒子数を増やして計算規模を大きくしても FFT2B の処理時間は変わらず、逆に全体に対する処理時間の占める割合は小さくなることから、FFT2B と USINM よりは、粒子数の

影響を受ける SRC, PSH1, PSH2 の高速化が重要である。

キャッシュ・ミスヒット時間は、インストラクションで 6.390 秒、オペランドで 44.759 秒である。インストラクション・キャッシュミスは全処理時間の 4.8%なので許容範囲とみる。オペランド・キャッシュミスは全処理時間の 33.6% である。これらはサブルーチン SRC,FFT2B,PSH1,PSH2 でほとんど発生している。DO ループ単位では、サブルーチン SRC の DO ループ 50 で 3%，と DO ループ 70 で 5%，サブルーチン PSH1 の DO ループ 20 で 4%，PSH2 の DO ループ 20 で 5%である。Figure 4.2.1～3 に各 DO ループを示す。サブルーチン PSH1 と PSH2 の DO ループ 20 は同じ構造である。

SRC の DO ループ 50 のキャッシュミスは、配列 G1XX, CZXX, LLAA のサイズが Table 4.2.2 に示すように大きく、さらに LLAA による間接番地アクセスとなっていることが原因である。SRC の DO ループ 70 は連続に配列をアクセスしているが、やはり配列のサイズが大きいためにキャッシュミスが発生していると考えられる。サブルーチン PSH2 の DO ループ 20 のキャッシュミスは、ELX のアクセスパターンが不規則であることとストライドが大きくなることが原因である。

Table 4.2.1 処理時間およびキャッシュミスヒット時間

Subroutine	CPU TIME		Inst. Cache Miss	Oper. Cache Miss		
TOTAL	133.112 sec.	100.0 %	6.390 sec.	4.8 %	44.759 sec.	33.6 %
SRC	36.377 sec.	27.6 %	0.035 sec.	0.0 %	11.833 sec.	9.0 %
FFT2B	27.620 sec.	20.7 %	4.621 sec.	3.0 %	9.166 sec.	7.0 %
PSH2	10.689 sec.	8.0 %	0.022 sec.	0.0 %	6.355 sec.	5.0 %
USINM	10.307 sec.	7.7 %	0.156 sec.	0.0 %	1.430 sec.	1.0 %
PSH1	9.601 sec.	7.2 %	0.020 sec.	0.0 %	5.793 sec.	4.0 %
FSINT	7.924 sec.	6.0 %	0.196 sec.	0.0 %	1.252 sec.	1.0 %
ATZFLD	5.117 sec.	3.8 %	0.402 sec.	0.0 %	1.380 sec.	1.0 %

```

DO 40 JJ = 1,NLENL / NDUP

DO 50 LL = 1,NVECL

  G1XX( LLAA(LL,JJ) ) = G1XX( LLAA(LL,JJ) ) + Q * ALFF(LL,JJ)
  CZEE( LLAA(LL,JJ) ) = CZEE( LLAA(LL,JJ) ) + Q * BLFF(LL,JJ)

50 CONTINUE

40 CONTINUE

```

Figure 4.2.1 サブルーチン SRC の DO ループ 50

```

DO 60 IM = 0,NDUP - 1

DO 70 IG = 1,NMAX

  G1X(IG) = G1X(IG) + G1XX(IG + NMAX * IM)
  CZE(IG) = CZE(IG) + CZEE(IG + NMAX * IM)

70 CONTINUE

60 CONTINUE

```

Figure 4.2.2 サブルーチン SRC の DO ループ 70

Table 4.2.2 DO ループ 50 の配列サイズ

配列名	型	サイズ	内 容
G1XX(NMAXL)	Real*8	8 バイト×NMAXL=233KB	メッシュ点上の電荷
CZEE(NMAXL)	Real*8	8 バイト×NMAXL=233KB	メッシュ点上の電流
LLAA(8,NLN)	Real*8	8 バイト×8×NLN=32KB	ワーク領域
ALFF(8,NLN)	Real*8	8 バイト×8×NLN=32KB	ワーク領域
BLFF(8,NLN)	Real*8	8 バイト×8×NLN=32KB	ワーク領域
G1X(NMAXL)	Real*8	8 バイト×NMAXL=233KB	メッシュ点上の電荷
CZE(NMAXL)	Real*8	8 バイト×NMAXL=233KB	メッシュ点上の電流

注) NLN=512. NMAXL = NNMAX * NDUP = NNCXPP * NNCYPP * NNCZ

= (NNCX + NNADX)*(NNCY + NNADY)*NNCZ=(32 + 11) * (32 + 11) * 16=29584.

```

DO 20 JJ = 1,NLEGL
I      = JFST + JJ
IF( Z(I) .NE. ZOUT ) THEN
  LX     = X(I) + 1.0
  LXR   = IR(LX)
  LY     = Y(I) + 1.0
  LYR   = JR(LY)
  LZ     = Z(I) + 1.0
  LZR   = KR(LZ)
  DX     = X(I) - LX + 1.
  DY     = Y(I) - LY + 1.
  DZ     = Z(I) - LZ + 1.
  .
  .
  .
C..... E.S. FIELD
EXT = ELX(LX ,LY ,LZ )*ALF1 + ELX(LXR,LY ,LZ )*ALF2
1      + ELX(LX ,LYR,LZ )*ALF3 + ELX(LXR,LYR,LZ )*ALF4
2      + ELX(LX ,LY ,LZR)*ALF5 + ELX(LXR,LY ,LZR)*ALF6
3      + ELX(LX ,LYR,LZR)*ALF7 + ELX(LXR,LYR,LZR)*ALF8
EYT = ELY(LX ,LY ,LZ )*ALF1 + ELY(LXR,LY ,LZ )*ALF2
1      + ELY(LX ,LYR,LZ )*ALF3 + ELY(LXR,LYR,LZ )*ALF4
  .

```

Figure 4.2.3 サブルーチン PSH2 の DO ループ 20

(2) 専用回路の適用可能性

まず、サブルーチン単位の解析結果について述べる。Table 4.2.3 に GYRO3D の主要ルーチンのベクトル処理性能を示す。GYRO3D の全体の処理時間は、スカラ処理；133.112 秒、ベクトル処理；12.811 秒、速度向上率；10.390、ベクトル化率；96.97%である。表に示したサブルーチンは、コード全体の処理時間に対してスカラ処理で 74%、ベクトル処理で 81.6%を占める。また、これらサブルーチンは、処理内容から次の 3 つに分類できる。

Particle Pushing 部： PSH1, PSH2

Field Solver 部： FFT2B, USINM, FSINT

Charge Deposit 部： SRC

ここで、PSH1, PSH2, SRC の処理時間は、粒子数に比例する。

PSH1, PSH2, SRC が 20 倍強の性能向上があるのにもかかわらず、処理全体の向上率は 10 倍程度となっている。これは、全体に占める割合がスカラで 20.7%である FFT2B のベクトル処理による性能向上が 3 倍程度しかないためである。今回の入力データでは FFT2B の全体におけるコストが高いが、粒子数を多くしても FFT2B の処理時間は、不变なので、FFT2B の全体におけるコストは低くなる。したがって、粒子数を多くすれば FFT2B の処理が隠れて、処理全体の性能向上率はもっと良くなる。また、PSH1, PSH2, SRC のベクトル処理については、配列へのアクセスが不連続（ランダム）であるため、ベクトル処理の性能を低下させている。

Table 4.2.3 GYRO3D の主要ルーチンのベクトル処理性能

	スカラ処理時間(S)	ベクトル処理時間(V)	速度向上率(S/V)
SRC	36.377 sec.	1.638 sec.	22.208
FFT2B	27.620 sec.	6.056 sec.	3.013
PSH2	10.689 sec.	0.436 sec.	24.516
USINM	10.307 sec.	0.659 sec.	15.640
PSH1	9.601 sec.	0.430 sec.	22.327
FSINT	7.924 sec.	0.228 sec.	34.754
ATZFLD	5.117 sec.	0.999 sec.	5.122

電子：4096 個、イオン：4096 個、高エネルギー粒子：1 個、メッシュ数：32×32×16 のデータを示す。

次に主要ループに対する解析結果について述べる。Table 4.2.4 に GYRO3D の主要ループのベクトル処理性能を示す。GYRO3D の場合、MD0 や MD1 と

違って、処理時間は多数のループに分散している。Table 4.2.3 には、CPU 時間が全体の 40%を占める 4 つのループのベクトル処理性能を示した。ここで、サブルーチン SRC の Do ループ 50 の平均ベクトル長は 128, Do ループ 70 の平均ベクトル長は 29,584, サブルーチン PSH1 及び PSH2 の Do ループ 20 の平均ベクトル長は 512 である。これらのループについて、専用回路化によりベクトル処理同様の高速化が見込める。その他のループについては、さらに検討が必要である。

Table 4.2.4 GYRO3D の主要ループのベクトル処理性能

	スカラ処理時間(S)	ベクトル処理時間(V)	速度向上率(S/V)
Loop 50 in subroutine SRC	6.443 sec.	0.275 sec.	23.430
Loop 70 in subroutine SRC	21.984 sec.	0.558 sec.	39.398
Loop 20 in subroutine PSH1	9.587 sec.	0.425 sec.	22.558
Loop 20 in subroutine PSH2	10.675 sec.	0.431 sec.	24.768

4.3 キャッシュ・チューニング

4.2 節で行ったオペランド・キャッシュミスの考察にもとづいて、キャッシュ・チューニングを行った。

(1) 概要

配列のソートによりアクセスパターンを最適化する。粒子番号 I に依存し物理量を示す配列 ELX(LX,LY,LZ)等のアクセスは不連続（三次元空間上でばらばら）であるため、キャッシュミスになる確率が高い。そのため、サフィックス LX, LY, LZ の大きさによって計算順序を変更する。すなわち、XYZ の三次元空間の粒子位置でソートし、できるだけ ELX に連続にアクセスすることで、キャッシュミスヒットを最小化する。ソートは肝心な数値計算以上にならないように粗いソート手順を使用する。ソート・アルゴリズムは、Direct Mapping を使用した。これをサブルーチン PSH2 の DO ループ 20 に適用した。PSH2 は本入力データでは処理時間が少ないが、粒子数の増加に比例して処理時間の増加するため、高速化が最も必要なサブルーチンである。

(2) チューニング結果

キャッシュチューニングによって、GYRO3D の処理時間とキャッシュ・ミスヒット時間は Table 4.3.1 のように改善された。DO ループ 20 のキャッシュ・チューニング前後のストライドの変化を Fig. 4.3.1 及び 4.3.2 に示す。チュー

ニングによってストライドが大幅に小さくなっていることがわかる。

Table 4.3.1 DO ループ 20 の処理時間とキャッシュ・ミスヒット時間の変化

	修正前	修正後
処理時間	10.675 sec.	7.836 sec.
キャッシュ・ミスヒット	6.351 sec.	3.682 sec.

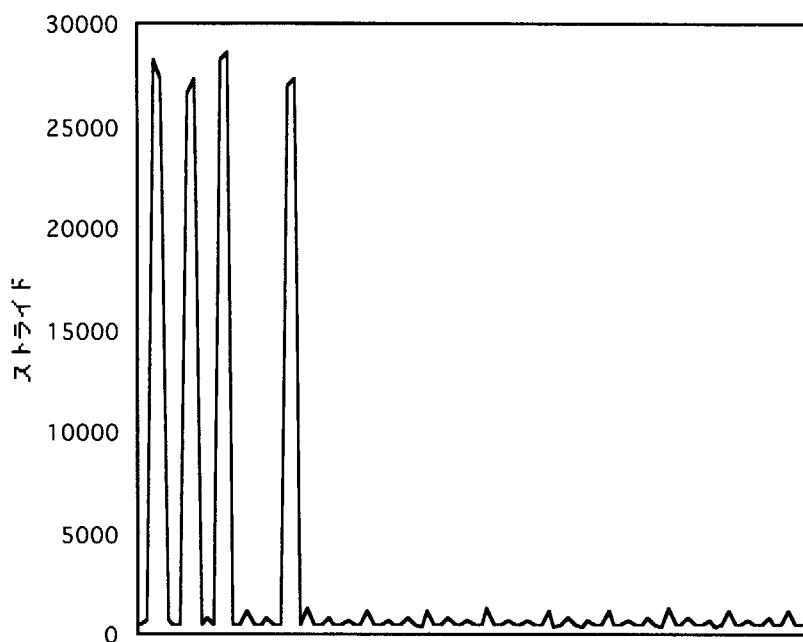


Figure 4.3.1 ストライドの変化（チューニング前）

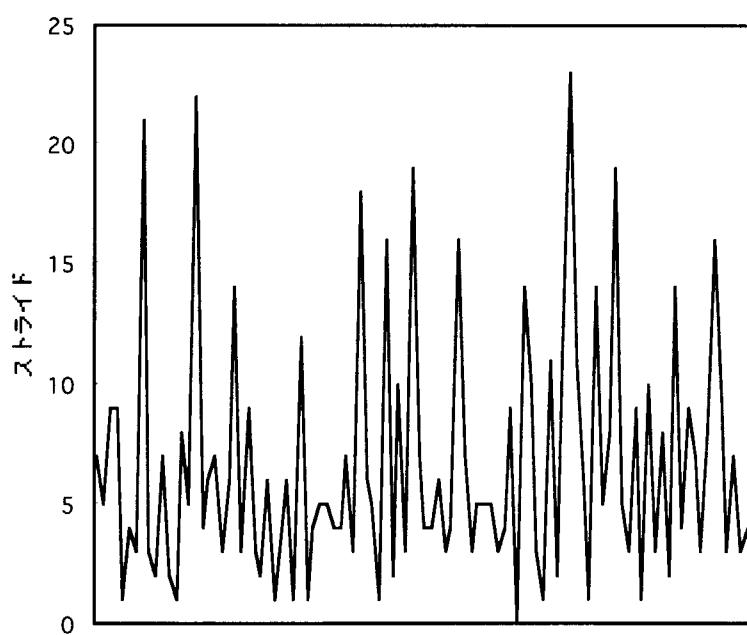


Figure 4.3.2 ストライドの変化（チューニング後）

Figure 4.3.3 に Direct Mapping 法によるソートのソース, Fig. 4.3.4 にキヤッショ・チューニング後の DO ループ 20 のソースを示す.

```

DO JJ=1,NTOT

INDEX(JJ)=0

ENDDO

KETSU=NTOT+1

XYPLANE=NCXPP*NCYPP

XFACT=(NTOT-1)/(XYPLANE*NCZ)

DO 11 JJ=1,NTOT

C      I=JFST+JJ

      I=JJ

      LZ=Z(I)+1.0d0

      LY=Y(I)+1.0d0

      XPOINT=XYPLANE*LZ+NCXPP*LY+X(I)

      IPOINT=XFACT*XPOINT

      IF(IPPOINT.GT.NTOT) IPOINT=NTOT

      IF(IPPOINT.LT.1) IPOINT=1

      IF(INDEX(IPPOINT).GE.4) GO TO 15

      INDEX(IPPOINT)=INDEX(IPPOINT)+1

      IXX=INDEX(IPPOINT)

      IQQQ(IXX,IPPOINT)=I

      GO TO 11

15    CONTINUE

```

Figure 4.3.3 Direct Mapping 法(1/2)

```
KETSU=KETSU-1  
ISORTED(KETSU)=I  
11  CONTINUE  
IPOINT=0  
DO 12 JJ=1,NTOT  
IXX=INDEX(JJ)  
IF(IXX.EQ.0) GO TO 12  
DO IDOT=1,IXX  
IPOINT=IPOINT+1  
ISORTED(IPOINT)=IQQQ(IDOT,JJ)  
ENDDO  
12  CONTINUE
```

Figure 4.3.3 Direct Mapping 法(2/2)

```

DO 20 JJ = 1,NLENGL
      JX      = JFST + JJ
      I = ISORTED(JX)
      IF( Z(I) .NE. ZOUT ) THEN
        LX      = X(I) + 1.0
        LXR     = IR(LX)
        LY      = Y(I) + 1.0
        LYR     = JR(LY)
        LZ      = Z(I) + 1.0
        LZR     = KR(LZ)
        DX      = X(I) - LX + 1.
        DY      = Y(I) - LY + 1.
        DZ      = Z(I) - LZ + 1.
        .
        .
        .
C..... E.S. FIELD
      EXT =   ELX(LX ,LY ,LZ )*ALF1 + ELX(LXR,LY ,LZ )*ALF2
      1      + ELX(LX ,LYR,LZ )*ALF3 + ELX(LXR,LYR,LZ )*ALF4
      2      + ELX(LX ,LY ,LZR)*ALF5 + ELX(LXR,LY ,LZR)*ALF6
      3      + ELX(LX ,LYR,LZR)*ALF7 + ELX(LXR,LYR,LZR)*ALF8
      EYT =   ELY(LX ,LY ,LZ )*ALF1 + ELY(LXR,LY ,LZ )*ALF2
      1      + ELY(LX ,LYR,LZ )*ALF3 + ELY(LXR,LYR,LZ )*ALF4
      .

```

Figure 4.3.4 DO ループ 20 (修正後)

5. 性能シミュレータ・プロトタイプの開発

次世代計算機の概念検討を定量的に進めるため、基本演算時間、計算機が装備する演算器数、キャッシュのヒット率、データのロード時間等計算機アーキテクチャに依存する因子を考慮しながら、処理時間を予測する性能シミュレータのプロトタイプを作成した。今回は、計算機については、RISC(Reduced Instruction Set Computer)型プロセッサを、計算コードについては、2章で述べた熱伝導一対流遷移シミュレーションコード MD0 を対象にシステムの試作及び妥当性評価を行った。

5.1 MD0 コードにおける計測箇所

MD0 コードを実行し、時間関数をソースに挿入する形式で実行時間及び、実行回数を計算した。計測を行った主要ルーチンについては次のとおり。

(1) force ルーチン:

粒子間力の計算をするルーチン。粒子間力の影響範囲のインデックステーブルをチェックし力の計算をしている。2重ループ (Do 270, 271) の構造になっているが、計算処理は内側の do ループに集中しているため 1重ループとして評価した。

(2) maxwell ルーチン

初期の速度分布（マックスウェル分布）を計算するルーチン。このルーチンは、実行において 1 回しか計算されない。また、粒子数の半分のループ長の do ループが 2 つあるが実行時間が短く、類似処理をしているので、1 ループとして評価した。

(3) propcel ルーチン

解析メッシュ上の物理量を粒子の速度などから計算している。Do ループは、8 個存在するが、1 つ 1 つの実行時間は短いので、実行回数の同じループ毎にまとめて 2 つのループとして評価。ループ長は、まとめたループでもっとも時間のかかっているループのループ長を採用した。

(4) pmoves ルーチン

解析領域内の粒子の移動と、境界近傍での境界処理。境界処理を行う 9300 のループはほとんど実行されない。解析領域内の処理をしている 300 のループの実行時間がルーチンの実行時間にほぼ等しい。したがって、do300 を代表させて 1 ループとして評価した。

(5) pcross ルーチン

境界領域を通過して解析領域外に出た粒子は、フラグを立て次の処理から除いている。1 ループのみであり、do10 ループの実行時間がルーチンの実行時間である。

(6) table ルーチン

粒子相互作用の及ぶと思われる範囲にある粒子についてテーブルを作成している。Force ルーチンで、このテーブルを使用している。この table ルーチンでは、全ての粒子間の距離を評価しているために実行時間が粒子数の 2 乗に比例している。ルーチンの実行時間は、最内のループの実行時間であるので 1 ループとして評価した。

(7) main プログラム

全体の計算制御。計算時間は、サブルーチンの呼び出しのためのオーバーヘッド及び、I/O 時間である。解析のメインループである 224 ループ内の 2 つのループと、224 ループ外の実行時間の 2 つのループと等価として評価した。実行時間は短い。

(8) uniran, chkbnd ルーチン

乱数の発生 (unirand) 及び、境界の通過のチェック (chkbnd) を行う。上記の親ルーチンのループの一部として（展開されたと仮定して）実行時間を評価した。

2 ケースの入力データに対する上記サブルーチン群の実行時間の総和とコード全体の実行時間の関係は次のとおり。

ケース 1 の場合：

Global Total = 5.7499E+01

Elapsed time = 5.7513E+01

ケース 2 の場合 :

Global Total = 3.4057E+01

Elapsed time = 3.4079E+01

ここで、 Global Total とは、 上記のサブルーチン毎に評価した実行時間の総和から時間計測用ルーチンのオーバーヘッド時間を差引いた時間である。 Elapsed time は、 MD0 コードの開始から終了までの実行時間である。両者は、 非常に近い値となっている。すなわち、 上記サブルーチン群が MD0 コードの 実行時間を代表していることがわかる。

5.2 性能予測における仮定

一般に、 コードの実行時間は、 次のように分類できる。

(1) データのロード時間及び、ストア時間

DO ループの場合は、 キャッシュヒット率が大きなファクタになる。

(2) 演算時間

演算には、 CPU 内部に演算器がある基本演算と、 それらの演算器を組み合 わせて計算する複合演算がある。

(3) call や return, goto によるオーバーヘッド時間 : Tov

(4) 入出力文による実行時間 : Tio

MD0 コードの場合、 実行時間の大小関係は、

$$(1) \geq (2) \gg (3) \geq (4)$$

のようになっている。 実際の実行時間は、 (1)と(2)の和の時間とほぼ等しい。 したがって、 (1), (2)の時間を評価すれば実際の実行時間は予測できる。 ここで、 現在の RISC プロセッサにおいては、 (1)及び、 (2)の実行は CPU 内部で並列に実行できるものと仮定した。また、 MD0 コードを DO ループの集合体と仮定して実行時間の予測を行った。

5.3 性能予測に使用したモデル

(1) データのロード及び、ストア時間: Tls

データのロード及びストア時間は、 メモリからキャッシュへ、 キャッシュ から CPU のレジスタへあるいは、 その逆に要する時間をさす。 キャッシュの

ヒット率やキャッシュからレジスタ、メモリからキャッシュへのデータ転送速度に大きく依存している。今、以下のように、 T_{ls} を定義する。

$$T_{ls} = (Nl + Ns * Fs) * Cf(n,x) * Tl$$

Tl : データのロードの平均時間

Nl : データのロード回数、右辺に現れる common 変数及び、配列変数の個数

ワーク変数は、レジスタに設定されると考えロード回数に入れないと。

Ns : データのストア回数、左辺に現れる common 変数及び、配列変数の個数

Fs : ロード時間を 1 とした時のストア時間の平均ファクタ。（1 に近い値）

$Cf(n,x)$: キャッシュミス影響予測子

ループの大きさ n と、キャッシュのヒット率 x に依存するファクタ。

キャッシュのヒット率の関数とし、実質上はループ長 n の関数として取り扱う。

(2) 演算時間 : Top

演算は、CPU 内部に演算器のある演算（以下、基本演算と呼ぶ）と、基本演算の組合せで実現されている演算（以下、複合演算と呼ぶ）に分類することができる。RISC プロセッサでは、基本演算は並列に実行されている。

- ・ 基本演算の種類 基本演算実行時間 Tpr : 実数, Tpi : 整数

Alpha, PA-RISC とも以下の演算器を持っている。

(a) + : 加算 - : 減算 * : 乗算 / : 除算 √ : 平方根

(b) 論理、比較演算

(c) 型変換 : 実数 \leftrightarrow 整数

ここで、(a)の + - * は 64 ビット長で完全パイプライン実行できる構造である。また、実数演算器の乗算器は独立に用意されており並列実行できる構造になっている。

- ・ 複合演算の種類 複合演算実行時間 Tc

基本演算の組合せで実現されている命令で、今回の解析の対象となった MD0 コードは、以下の演算を含んでいる。

** : べき算 (**2 **4 などは掛け算に翻訳される可能性有り)

sin,cos : 三角関数, log : 対数関数

mod : 剰余, max,min : 最大最小

rand : 一様乱数の生成

これらの基本関数は、ライブラリ関数として RISC プロセッサの機能を最大限引き出すようにコーディングされている。実行速度はライブラリの性能に依存すると考えられる。

これらのことと踏まえて、演算にかかる全体の処理時間 : Top を次のようにモデル化した。

$$Top = Tpr + Tpi + Tc$$

$$Tpr = \sum Npr(i) * Fpr(i) \quad i: \text{実数基本演算部分}$$

$Npr(i)$: 基本演算の出現個数

$Fpr(i)$: 時間ファクタ

$$Tpi = \sum Npi(i) * Fpi(i) \quad i: \text{整数基本演算部分}$$

$$Tc = \sum Nc(i) * Fc(i) \quad i: \text{複合演算部分}$$

さらに、Do ループに対する実行時間 : Tdo を、ロード／ストアと演算が平行実行されることを考慮し、以下の式で評価する。

$$Tdo = \max[(Nl + Ns * Fs) * Cf(N, x) * T1 / Pe_do(N), Top / Pe_do(N)] * N$$

ここで、N はループ長、Pe_do(N) は、CPU 内部の並列度を現わす比例係数である。上式の意味するところは

ループ実行時間 = $\max[(\text{ロード/ストアのコスト}) * \text{ループ長} * \text{キャッシュミス率}, (\text{演算コスト}) * \text{ループ長}]$

である。

$Cf(N, x)$ については、[ロード/ストアのコスト]*N 及び、[演算コスト]*N を計測後、Pe_do をハードウェアスペックから決定すると、計算することができる。Pe_do は、今回、すべての DO ループで 1.0 を与えた。したがって、Cf は、

$$C_f(N,x) = T_{DO} / ([ロード/ストアのコスト]*N)$$

で計算することができる。複数のデータで測定したところ、 C_f は DO ループごとにほぼ一定値を取ることが分かった。

5.4 評価結果

今回作成した性能予測モデルを MD0 コードに適用し、予測結果の妥当性を評価するために実機マシンを使って性能測定を行った。すなわち、予測方程式の係数の計算と、パラメータを変化させた時の予測方程式の妥当性を評価した。

現在、シミュレータは Alpha21264(667MHz)チップの性能を基に作成されている。異なったマシンで性能を評価する場合は、シミュレータに組み込まれた基本性能に関するデータベース部分を書き換える必要がある。

評価に使用した計算機は、ビジュアルテクノロジー社製の計算機で、スペックは以下の通りである。

名称	VT-Alpha6 シリーズ
CPU	Alpha21264 667MHz
1 次キャッシュ	128KB (オンチップ)
2 次キャッシュ	4MB (オンボード)
メモリバンド幅	最大転送速度 2.6GB/秒
搭載メモリ	1GB
OS	Tru64 UNIX V4.0

(1) キャッシュミス影響予測子(C_f)の評価

データがすべてキャッシュに存在する場合、データのロード/ストアに要する時間は、キャッシュからのロード/ストアに要する時間とロード/ストアの回数の積で求められる。しかし、実際は必ずしもすべてのデータがキャッシュに存在するとは限らず、キャッシュミスが発生するとロード/ストアに要する時間は大幅に増加する。これまで述べたように、このキャッシュミスによってロード/ストアに要する時間が増加する割合をキャッシュミス影響予測子 C_f として性能予測モデルに取り込んだ。 C_f はキャッシュのヒット率を 100%として予測した解析時間と実機テストにおける測定値を比較することで決定した。すなわち、キャッシュミス影響予測子 C_f の値を評価するため、

9 ケースのデータに対して、実測値と性能予測モデルによる予測値と比較した。

この際、MD0 コードの実行に用いたパラメータは以下のとおりである。

粒子数密度	0.3	下面境界温度	120K
アスペクト比	2.016	解析時間ステップ	0.01 秒
RINT 4	I.snap	12	
Cuf Off 距離	3	Itimax	20
上面境界温度	1200K	評価ステップ間隔	20

上記パラメータを用いて MD0 コードをセル数と粒子数を変えた 9 ケースに対し実行時間を予測／計測した結果は次のとおり。

セル数	粒子数	衝突セル数	予測値(秒)	計測値(秒)	予測/計測
80x100	72000	3	9.531E+01	9.391E+01	1.015
80x120	86400	3	1.251E+02	1.212E+02	1.032
80x140	100800	3	1.585E+02	1.568E+02	1.011
100x160	144000	3	2.800E+02	2.775E+02	1.009
160x160	230400	3	6.195E+02	6.301E+02	1.001
80x100	128000	4	2.303E+02	2.314E+02	0.983
80x120	153600	4	3.102E+02	3.156E+02	0.983
80x140	179200	4	4.015E+02	4.016E+02	1.000
100x160	256000	4	7.430E+02	7.611E+02	0.976

この結果を用いて主要な DO ループ（サブルーチン）ごとに Cf の値を計算した。

force	Cf = 8.53E-01
maxwell	Cf = 3.03E+01
propcel 1	Cf = 2.71E+00
2	Cf = 6.00E+01
pmoves	Cf = 4.69E+00
pcross	Cf = 6.00E+00
table	Cf = 2.87E+00
main 1	Cf = 2.86E+02
2	Cf = 2.50E+00

なお、実測値と C_f のグラフを付録 D に示す。 C_f は、ループ長 N とキャッシュ・ヒット率 x の関数であるが、今回の作業では、平均化した値を用いて実行時間の予測を行っている。

(2) C_f 補正後の結果

以下は、求めたキャッシュミス影響予測子 C_f を用いて、パラメータとしてセル数を変更して解析プログラムの予測値と実際の実測値を比較したものである。ループ長が小さい 2 ケースと、大きな 1 ケースに対して計測と予測を行った。なお、セル数以外のパラメータは(1)と同じで、衝突セル数は 4 である。

セル数	粒子数	予測値(秒)	計測値(秒)	予測 / 計測
1) 50x 50	40000	41.63	41.51	1.00
2) 60x 60	57600	68.68	70.69	0.97
3) 110x160	281600	879.37	897.86	0.98

実測値と予測値が良く一致している事が分かる。

次に、MD0 の 解析パラメータを変化させた場合の実機における測定から計算した解析時間の予測と実際に計測した結果を示す。

文献[11]では、上面と下面の温度差をパラメータにした対流遷移についての計算結果が報告されているが、シミュレーションの実行時間の評価では、上面下面の温度差は、粒子インデックスの評価サイクルにのみ影響する。これは、table ルーチンを呼出す回数に影響する。したがって、table ルーチンの評価が妥当であれば、温度差の実行時間への影響は正しく評価できることを意味する。同様のことが、アスペクト比、解析時間ステップについても当てはまる。

今回は、最も実行時間がかかるている table、及び force ルーチンの実行時間に影響する粒子密度、RINT 及び、Cut Off 距離を変化させて、計測時間と予測時間を比べてみた。

基準となる値は、最初から Md0 コードに埋め込まれていた

粒子数密度 : 0.3

RINT : 4.0
 Cut Off : 3.0

である。またメッシュは、 80×100 、粒子数 72000、衝突セル数 3 とする。
 基準データでの実行時間は計測値 93.91 秒、予測値 95.31 秒である。

(a) 粒子数密度

	粒子数密度	予測値(秒)	計測値(秒)	予測 / 計測
(1)	0.2	81.75	88.64	0.92
(2)	0.4	108.86	111.86	0.97
(3)	0.5	122.41	126.50	0.97
(4)	0.6	135.97	135.62	1.00

(b) RINT 変数

	RINT	予測値(秒)	計測値(秒)	予測 / 計測
(1)	3.0	78.18	87.08	0.90
(2)	5.0	117.33	119.12	0.98
(3)	6.0	144.25	139.76	1.03
(4)	7.0	176.07	165.83	1.06

(c) Cut Off 変数

	Cut Off	予測値(秒)	計測値(秒)	予測 / 計測
(1)	2.0	72.72	79.81	0.91
(2)	4.0	126.93	128.42	0.99
(3)	5.0	167.59	161.71	1.04
(4)	6.0	217.29	209.09	1.04

予測値と計測値は良く一致している。これから table 及び、force ルーチンのコストの見積もりが正しいことが結論づけられる。

(3) 性能シミュレータの利用法

性能シミュレータの利用方法を付録 E に示す。エクセルを利用して熱伝導-対流遷移コード (md0) の性能シミュレータの実行に必要なパラメータを入力する。入力は、直接エクセルのシート上で行い、予測結果を自動的に計算し、表示させるようにした。入力可能セルは、青字で表示しているセルで、計算結果は、赤字で表示している。

シートは保護されており、青字の部分以外は入力不可であるが、パスワードは設定していないので保護の解除で全ての領域で入力可能となる。具体的には、次のパラメタを設定し、実行時間を評価する。

- ・ 解析条件パラメタ

- (a) Number Dencity = 0.4
- (b) アスペクト比 = 2.016

(アスペクト比、(1)と粒子数 n から width,hight を計算している)

- (c) rint = 4.0
- (d) Cut Off = 3.0
- (e) 上面温度 = 1200.0
- (f) 下面温度 = 120.0

境界温度が粒子相互作用のインデックステーブルを評価する
タイムステップ間隔を決めている。

- (g) 解析時間ステップ = 0.01

- ・ セル情報

- (h) x 方向セル数 = 40
- (i) y 方向セル数 = 40
- (j) 1 セル当りの粒子数 = 4 (平方数でないといけない。4,9,16,25....)

- ・ 時間ステップ情報

- (k) isnap = 6
- (l) itimax = 20
- (m) サンプルステップ間隔 = 20

6.まとめ

特殊演算回路を装備した汎用 MPU を要素計算機とする並列計算機を想定して次世代計算機の概念検討を進めた。ここでは、演算性能とデータ供給性能の二つの観点から、量子計算コードを高速化するためのハードウェアの概念検討を進めた。具体的には、専用回路を適用する計算処理を特定するためのコードの挙動解析と専用回路を適用した場合の性能予測を行うためのシミュレータの開発に着手した。

6.1 量子計算コードの挙動解析

次世代計算機によって高速化する分野については、流体計算については地球シミュレータが、また、粒子計算のうちモンテカルロ・コードについては既に高速モンテカルロ装置が適合することから、粒子輸送モンテカルロ・コード以外の粒子計算を対象に概念検討を進めた。すなわち、熱伝導一対流遷移コード MD0、溶融塩粘性率計算コード MD1、三次元ジャイロ運動論粒子コード GYRO3D を対象に挙動解析を行った。

MD0 コードの場合、サブルーチン Table 中の DO ループ 100 及びサブルーチン Force 中の DO ループ 271 に全体の 95% の計算コストが集中している。ベクトル処理性能との比較から、これら二つの DO ループについて、専用回路化によりベクトル処理同等以上の高速化が見込める。

MD1 コードの場合、DO ループ 7000, 96, 199, 97 に全体の 97% の計算コストが集中している。これらすべての Do ループについて、専用回路化によりベクトル処理同等以上の高速化が見込める。

GYRO3D コードの場合、MD0 や MD1 と違って、処理時間は多数のループに分散している。Table 4.2.3 に示した、全体の 40% を占める 4 つのループについては、専用回路化によりベクトル処理同等以上の高速化が見込める。その他のループについては、さらに検討が必要である。

6.2 性能シミュレータ

今回、MD0 コードを対象として試作した、性能シミュレータは、ロード／ストア及び演算のコスト（処理に要する CPU 時間の単価）を計算するものである。シミュレータの特性としてコンパイラが行うような正確な構文解析によるコスト評価を行わなくても、実行時間が予測・評価できることを確かめ

た。さらに、予測値と実機テスト結果とほぼ一致していることが確認できた。

性能予測モデルについては、キャッシュのミス/ヒットの定式化の第一段階としてキャッシュミス影響予測子(C_f)を検討した。その結果、キャッシュの有効利用ができ、データの再利用性が高い場合は、理論値と同等の性能となることが確認できた。キャッシュミスが発生する場合は数十倍、キャッシュ上にデータが無くメモリアクセスによってデータロードする場合は数百倍、キャッシュヒットする場合の理論性能値よりも遅くなるという現象を定量化することができた。

6.3 今後の課題

コードの挙動解析については、今回、代表的な3つの量子計算コードについて解析を行い、専用回路の適用範囲とその効果を検討したものの、一般性に関する検討が不十分であった。量子計算分野における汎用性を考慮した計算処理の特定が今後の課題となる。

シミュレータの開発においては、性能予測モデルとソースコードの解析プログラムを開発し、コンピュータの基本性能が不明な場合にも対応できるよう的基本性能測定プログラムも作成した。これによって、性能予測シミュレーションに必要となる要素技術を確立することができた。今後は、性能を決定する要因の一つであるキャッシュの効果をより詳細に定量化する方法論の確立が必要である。具体的には、キャッシュ・サイズやライン数などのハードウェア特性をパラメータ化し、キャッシュ・ヒット率をプログラムの処理流れから算出するツールの開発が必要となる。これによって、今回作成した性能予測モデルのキャッシュミス影響予測子(C_f)の精度を上げができる。すなわち、性能予測精度の向上と汎用化を図ることが可能である。

参考文献

- [1] Higuchi, K., Asai, K. and Akimoto, M. : Development of Monte Carlo Machine for Particle Transport Problem, J. Nucl. Sci. Technol., 32, 10(1995).
- [2] 細見 岳生, 森 真一郎, 中島 浩, 富田 真治 : ディレクトリ型キャッシュコヒーレンスプロトコルの性能評価, 情報処理学会論文誌, Vol. 37, No. 2(1996).
- [3] Eric Rotenberg, Steve Bennett, and James E. Smith : A Trace Cache Microarchitecture and Evaluation, IEEE Trans. Comput., Vol. 48, No. 2, pp. 111-120(1999).
- [4] Doug Joseph and Dirk Grunwald: Prefetching Using Markov Predictors, IEEE Trans. Comput., Vol.48, No.2, pp.121-133(1999).
- [5] Wayne A. Wickelgren : Single-Trace Fragility Theory of Memory Dynamics, Memory & Cognition, Vol. 2, No.4, pp.775-780(1974).
- [6] 藤井 実, 浅井 清 : 階層的ファイル自動管理システムの設計, 情報処理学会論文誌, Vol.21, No.6, pp.443-453(1980).
- [7] 渡辺 正, 薫木 英雄 : 粒子法による熱伝導-対流遷移の研究, JAERI-Research 96-046 (1996).
- [8] 神山新一, 佐藤明 : 分子動力学シミュレーション, 朝倉書店 (1997) .
- [9] SUPER-UX ANALYZER-P/SX 利用の手引き, G1AF15-2, 日本電気株式会社 (1995).
- [10] 寒川光 : RISC 超高速化プログラミング技法, 共立出版 (1995).
- [11] 岡本 芳浩, 横川 三津夫, 小川 徹 : 分子動力学法による溶融塩の粘性率の計算, JAERI-M 93-242 (1993).
- [12] Hiroshi Naito, Takuya Sonoda, Shinji Tokuda, Viktor K. Decyk : Parallelization of Gyrokinetic Particle Code and its Application to Internal Kink Mode Simulation, プラズマ・核融合学会誌, Vol.72, No.3 (1995).

付録 A MD0 の ANALYZER-P による解析結果

(a) SUMMARY LIST

ANALYZER-P/SX REVISION : REV.260
 ANALYZED DATE : Mon Aug 23 17:19:27 1999
 FANP OPTIONS : -tm -amm -AL f=md0-tm.anaout
 : lpunit all programs
 -A0
 : do all programs
 -m1

 EXECUTION TIME : CPU TIME = 0 : 12 ' 14 " 805 (734.805 sec)
 ELAPSED TIME = 0 : 16 ' 20 " 842 (980.842 sec)

 INSTRUCTION INFORMATION: INSTRUCTION COUNT = 54662313053
 FLOATING POINT ELEMENT COUNT = 23924782662
 VECTOR INSTRUCTION COUNT = 480302
 VECTOR ELEMENT COUNT = 60970496

 PERFORMANCE : 74.473 MOPS 32.559 MFLOPS (BY CPU TIME)

 MEMORY INFORMATION : BANK CONFLICT TIME(sec) = 0.000(0.0 %)
 INST. CACHE MISS(sec) = 18.933(2.6 %)
 OPER. CACHE MISS(sec) = 78.046(10.6 %)

 VECTOR INFORMATION : VECTOR OPERATION RATIO = 0.11 %
 AVERAGE VECTOR LENGTH = 126.9

(b) PROGRAM UNIT SUMMARY LIST

PROG.UNIT	ATR.	CODE	FREQUENCY	INCLUSIVE		EXCLUSIVE		MOPS	MFLOPS	V.OP.	AVER.	MEMORY LOSS TIME			
				CPU TIME(%)	CPU TIME(%)	RATIO	V.LEN					BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)	
TABLE	SUB		1	522.556(71.1)	521.822(71.0)	79.8	36.8	0.00	46.7	0.000(0)	0.758(0)	61.575(8)			
FORCE	SUB		121	174.051(23.7)	174.049(23.7)	58.2	24.9	0.00	0.0	0.000(0)	7.477(1)	12.815(2)			
CHKBD	SUB	9600030		12.018(1.6)	12.018(1.6)	62.5	5.6	0.00	0.0	0.000(0)	7.962(1)	0.292(0)			
PCROSS	SUB		120	23.094(3.1)	11.074(1.5)	60.2	4.3	0.00	0.0	0.000(0)	1.078(0)	0.931(0)			
PROPCEL	SUB		120	6.637(0.9)	6.629(0.9)	114.6	12.3	0.00	68.0	0.000(0)	0.362(0)	0.655(0)			

MAIN	MAIN	1	734.805(100.0)	5.224(0.7)	83.5	19.0	0.00	32.1	0.000(0)	0.358(0)	0.718(0)
PMOVES	SUB	120	2.460(0.3)	2.176(0.3)	110.3	39.7	0.00	0.0	0.000(0)	0.003(0)	0.619(0)
GETTOD	SUB	82088	0.750(0.1)	0.750(0.1)	12.4	0.1	0.00	0.0	0.000(0)	0.597(0)	0.367(0)
FJRAND	SUB	240000	0.709(0.1)	0.709(0.1)	122.5	0.3	70.26	127.0	0.000(0)	0.259(0)	0.066(0)
MAXWELL	SUB	1	0.748(0.1)	0.193(0.0)	93.6	35.1	0.00	0.0	0.000(0)	0.000(0)	0.004(0)
UNIRAN	SUB	4	0.837(0.1)	0.128(0.0)	17.7	3.7	0.03	127.0	0.000(0)	0.075(0)	0.004(0)
UNINIT	ENT		1								
INIT	SUB	1	0.032(0.0)	0.032(0.0)	64.6	14.9	0.00	0.0	0.000(0)	0.004(0)	0.000(0)

(c) PROGRAM UNIT DETAIL LIST (TABLE)

LINE	NESTING	FREQUENCY INCLUSIVE			EXCLUSIVE			AVERAGE P/V	AVERAGE	MEMORY LOSS TIME		
		CPU TIME(%)	CPU TIME(%)	CPU TIME	CPU TIME(%)	CPU TIME	CPU TIME			LOOP.LEN. BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)
000001	TABLE	1	522.556(71.1)	0.000(0.0)	522.556			0.000(0)	0.000(0)	0.000(0)		
000010	-GETTOD	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)		
000029--000047 -DO 101		1	522.555(71.1)	1.390(0.2)	522.555	79999	0.000(0)	0.415(0)	0.491(0)			
000027--000040 --DO 100		79999	520.432(70.8)	520.432(70.8)	0.007	40000	0.000(0)	0.342(0)	61.084(8)			
000045	--GETTOD	79999	0.734(0.1)	0.734(0.1)	0.000			0.000(0)	0.583(0)	0.360(0)		
000051	-GETTOD	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)		
000054	-GETTOD	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)		

(d) PROGRAM UNIT DETAIL LIST (FORCE)

LINE	NESTING	FREQUENCY INCLUSIVE			EXCLUSIVE			AVERAGE P/V	AVERAGE	MEMORY LOSS TIME		
		CPU TIME(%)	CPU TIME(%)	CPU TIME	CPU TIME(%)	CPU TIME	CPU TIME			LOOP.LEN. BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)
000001	FORCE	121	174.051(23.7)	0.003(0.0)	1.438			0.000(0)	0.001(0)	0.001(0)		
000010	-GETTOD	121	0.001(0.0)	0.001(0.0)	0.000			0.000(0)	0.001(0)	0.000(0)		
000014--000017 -DO 225		121	0.233(0.0)	0.233(0.0)	0.002	80000	0.000(0)	0.000(0)	0.000(0)			
000021--000083 -DO 270		121	173.814(23.7)	12.648(1.7)	1.436	79999	0.000(0)	4.050(1)	1.233(0)			
000031--000073 --DO 271		9679879	161.166(21.9)	161.166(21.9)	0.008	67	0.000(0)	3.426(0)	11.581(2)			
000086	-GETTOD	121	0.001(0.0)	0.001(0.0)	0.000			0.000(0)	0.001(0)	0.001(0)		

(e) PROGRAM UNIT DETAIL LIST (MAIN)

LINE	NESTING	FREQUENCY	INCLUSIVE	EXCLUSIVE	AVERAGE	P/V	AVERAGE	MEMORY LOSS TIME					
								CPU TIME(%)	CPU TIME(%)	CPU TIME	LOOP.LEN. BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)
000001	MAIN	1	734.805(100.0)	2.713(0.4)	734.805			0.000(0)	0.344(0)	0.012(0)			
000007	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000015--000017	-DO	1	0.000(0.0)	0.000(0.0)	0.000			20	0.000(0)	0.000(0)	0.000(0)		
000112	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000114	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000116	-UNINIT/UNIRAN/	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000121	-INIT	1	0.032(0.8)	0.032(0.8)	0.032			0.000(0)	0.004(0)	0.000(0)			
000122	-TABLE	1	522.556(71.1)	521.822(71.0)	522.556			0.000(0)	0.758(0)	61.575(8)			
000124	-MAXWELL	1	0.748(0.1)	0.193(0.0)	0.748			0.000(0)	0.000(0)	0.004(0)			
000125	-FORCE	1	1.434(0.2)	1.434(0.2)	1.434			0.000(0)	0.061(0)	0.109(0)			
000129	-TABLE	0	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000135--000193	-DO 224	1	207.322(28.2)	0.000(0.0)	207.322			6	0.000(0)	0.000(0)	0.000(0)		
000139--000191	--DO 200	6	207.322(28.2)	0.031(0.0)	34.554			20	0.000(0)	0.013(0)	0.005(0)		
000143	--PCROSS	120	23.094(3.1)	11.074(1.5)	0.192			0.000(0)	1.078(0)	0.931(0)			
000144	--PMOVES	120	2.460(0.3)	2.176(0.3)	0.020			0.000(0)	0.003(0)	0.619(0)			
000146	--TABLE	0	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000148	--GETTOO	120	0.001(0.0)	0.001(0.0)	0.000			0.000(0)	0.001(0)	0.000(0)			
000149--000153	--DO 220	120	1.225(0.2)	1.225(0.2)	0.010			80000	0.000(0)	0.001(0)	0.346(0)		
000154	--GETTOO	120	0.001(0.0)	0.001(0.0)	0.000			0.000(0)	0.001(0)	0.000(0)			
000156	--FORCE	120	172.616(23.5)	172.615(23.5)	1.438			0.000(0)	7.417(1)	12.706(2)			
000158	--GETTOO	120	0.001(0.0)	0.001(0.0)	0.000			0.000(0)	0.001(0)	0.000(0)			
000159--000163	--DO 300	120	1.225(0.2)	1.225(0.2)	0.010			80000	0.000(0)	0.001(0)	0.346(0)		
000164	--GETTOO	120	0.001(0.0)	0.001(0.0)	0.000			0.000(0)	0.001(0)	0.000(0)			
000180--000183	--DO 305	6	0.031(0.0)	0.031(0.0)	0.005			80000	0.000(0)	0.000(0)	0.008(0)		
000189	--PROPCEL	120	6.637(0.9)	6.629(0.9)	0.055			0.000(0)	0.362(0)	0.655(0)			
000195	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000197	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000199	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000210	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			
000212	-GETTOO	1	0.000(0.0)	0.000(0.0)	0.000			0.000(0)	0.000(0)	0.000(0)			

付録 B MDI の ANALYZER-P による解析結果

(a) SUMMARY LIST

```

ANALYZER-P/SX REVISION : REV.260
ANALYZED DATE       : Fri Mar  5 11:23:30 1999
FANP OPTIONS        : -tm -amm -AL f=md12.anaout
                        : lpunit all programs
                        : -AO
                        : do all programs
                        : -ml

EXECUTION TIME      : CPU TIME      = 1 : 27 ' 39 " 120 ( 5259.120 sec)
                      ELAPSED TIME = 1 : 29 ' 28 " 816 ( 5368.816 sec)

INSTRUCTION INFORMATION :
  INSTRUCTION COUNT          = 506386251115
  FLOATING POINT ELEMENT COUNT = 137472807302
  VECTOR INSTRUCTION COUNT   = 6
  VECTOR ELEMENT COUNT       = 154

PERFORMANCE          : 96.287 MOPS    26.140 MFLOPS (BY CPU TIME)

MEMORY INFORMATION   :
  BANK CONFLICT TIME(sec) = 0.000( 0.0 %)
  INST. CACHE MISS(sec)  = 6.026( 0.1 %)
  OPER. CACHE MISS(sec) = 391.557( 7.4 %)

VECTOR INFORMATION   :
  VECTOR OPERATION RATIO = 0.00 %
  AVERAGE VECTOR LENGTH = 25.7

```

(b) PROGRAM UNIT SUMMARY LIST

PROG.UNIT	ATR.	CODE	FREQUENCY	INCLUSIVE		EXCLUSIVE		MOPS	MFLOPS	V.OP. AVER.	MEMORY LOSS TIME		
				CPU TIME(%)	CPU TIME(%)	RATIO	V.LEN				BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)
MATN_	MAIN		1	5259.120(100.0)	5259.120(100.0)	96.3	26.1	0.00	25.7	0.000(0)	6.026(0)	391.557(7)	
ERFC	FUNC		0	0.000(0.0)	0.000(0.0)	0.0	0.0	0.00	0.0	0.000(0)	0.000(0)	0.000(0)	

(c) PROGRAM UNIT DETAIL LIST (MAIN_)

LINE	NESTING	FREQUENCY	INCLUSIVE	EXCLUSIVE	AVERAGE	P/V	AVERAGE	MEMORY LOSS TIME			
								CPU TIME(%)	CPU TIME(%)	CPU TIME	LOOP.LEN.
											BANK CONF.(%)
000001	MAIN_	1	5259.120(100.0)	0.836(0.0)	5259.120			0.000(0)	0.133(0)	0.011(0)	
000219--000221 -DO 203		1	0.000(0.0)	0.000(0.0)	0.000			4096	0.000(0)	0.000(0)	0.000(0)
000223--000231 -DO 201		1	0.000(0.0)	0.000(0.0)	0.000			2	0.000(0)	0.000(0)	0.000(0)
000227--000229 --DO 200		2	0.000(0.0)	0.000(0.0)	0.000			2048	0.000(0)	0.000(0)	0.000(0)
000233--000235 -DO 202		1	0.000(0.0)	0.000(0.0)	0.000			4096	0.000(0)	0.000(0)	0.000(0)
000306--000328 -DO 500		1	0.016(0.0)	0.016(0.0)	0.016			3000	0.000(0)	0.001(0)	0.000(0)
000337--000350 -DO 250		1	0.000(0.0)	0.000(0.0)	0.000			17	0.000(0)	0.000(0)	0.000(0)
000338--000350 --DO 250		17	0.000(0.0)	0.000(0.0)	0.000			17	0.000(0)	0.000(0)	0.000(0)
000339--000350 ---DO 250		289	0.000(0.0)	0.000(0.0)	0.000			17	0.000(0)	0.000(0)	0.000(0)
000355--000359 -DO 600		1	0.000(0.0)	0.000(0.0)	0.000			2	0.000(0)	0.000(0)	0.000(0)
000356--000359 --DO 600		2	0.000(0.0)	0.000(0.0)	0.000			80	0.000(0)	0.000(0)	0.000(0)
000361--000363 -DO 100		1	0.000(0.0)	0.000(0.0)	0.000			2	0.000(0)	0.000(0)	0.000(0)
000369--000371 -DO 150		1	0.000(0.0)	0.000(0.0)	0.000			2	0.000(0)	0.000(0)	0.000(0)
000385--000388 -DO 700		1	0.000(0.0)	0.000(0.0)	0.000			3	0.000(0)	0.000(0)	0.000(0)
000386--000388 --DO 700		3	0.000(0.0)	0.000(0.0)	0.000			1400	0.000(0)	0.000(0)	0.000(0)
000395--000401 -DO 301		1	0.001(0.0)	0.000(0.0)	0.001			3	0.000(0)	0.000(0)	0.000(0)
000397--000399 --DO 300		3	0.001(0.0)	0.001(0.0)	0.000			4096	0.000(0)	0.000(0)	0.000(0)
000403--000409 -DO 303		1	0.001(0.0)	0.000(0.0)	0.001			3	0.000(0)	0.000(0)	0.000(0)
000405--000407 --DO 302		3	0.001(0.0)	0.001(0.0)	0.000			4096	0.000(0)	0.000(0)	0.000(0)
000423--000431 -DO 1002		1	0.003(0.0)	0.000(0.0)	0.003			3	0.000(0)	0.000(0)	0.000(0)
000425--000429 --DO 1000		3	0.003(0.0)	0.003(0.0)	0.001			4096	0.000(0)	0.000(0)	0.000(0)
000438--000446 -DO 1021		1	0.001(0.0)	0.000(0.0)	0.001			3	0.000(0)	0.000(0)	0.000(0)
000440--000444 --DO 1011		3	0.001(0.0)	0.001(0.0)	0.000			4096	0.000(0)	0.000(0)	0.000(0)
000453--000455 -DO 1010		1	0.000(0.0)	0.000(0.0)	0.000			2	0.000(0)	0.000(0)	0.000(0)
000456--000462 -DO 1102		1	0.000(0.0)	0.000(0.0)	0.000			3	0.000(0)	0.000(0)	0.000(0)
000458--000460 --DO 1100		3	0.000(0.0)	0.000(0.0)	0.000			2048	0.000(0)	0.000(0)	0.000(0)
000463--000469 -DO 1103		1	0.000(0.0)	0.000(0.0)	0.000			3	0.000(0)	0.000(0)	0.000(0)
000465--000467 --DO 1101		3	0.000(0.0)	0.000(0.0)	0.000			2048	0.000(0)	0.000(0)	0.000(0)
000474--000477 -DO 1300		1	0.000(0.0)	0.000(0.0)	0.000			2	0.000(0)	0.000(0)	0.000(0)
000483--000498 -DO 1401		1	0.002(0.0)	0.000(0.0)	0.002			3	0.000(0)	0.000(0)	0.000(0)
000485--000488 --DO 1400		3	0.002(0.0)	0.002(0.0)	0.001			4096	0.000(0)	0.000(0)	0.000(0)
000496--000498 -DO 1412		1	0.000(0.0)	0.000(0.0)	0.000			2048	0.000(0)	0.000(0)	0.000(0)
000501--000503 -DO 1414		1	0.000(0.0)	0.000(0.0)	0.000			2048	0.000(0)	0.000(0)	0.000(0)
000527--000541 -DO 38	200	0.332(0.0)	0.001(0.0)	0.002			3	0.000(0)	0.000(0)	0.000(0)	
000529--000539 --DO 39	600	0.331(0.0)	0.331(0.0)	0.001			4096	0.000(0)	0.001(0)	0.018(0)	

000546--000552 -DO 94	200	0.059(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000548--000550 -DO 93	600	0.059(0.0)	0.059(0.0)	0.000	4096	0.000(0)	0.000(0)	0.000(0)
000555--000558 -DO 10	200	0.059(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000556--000558 -DO 10	600	0.059(0.0)	0.059(0.0)	0.000	4096	0.000(0)	0.000(0)	0.000(0)
000566--000616 -DO 98	200	1240.909(23.6)	2.195(0.0)	6.205	618	0.000(0)	0.319(0)	1.081(0)
000572--000578 -DO 97	123600	270.815(5.1)	270.815(5.1)	0.002	2048	0.000(0)	0.385(0)	10.049(0)
000581--000587 -DO 199	123600	274.267(5.2)	274.267(5.2)	0.002	2048	0.000(0)	0.408(0)	4.587(0)
000593--000596 --DO 8000	123600	51.765(1.0)	51.765(1.0)	0.000	4096	0.000(0)	0.112(0)	4.316(0)
000600--000607 -DO 96	123600	556.573(10.6)	556.573(10.6)	0.005	4096	0.000(0)	0.313(0)	158.299(3)
000611--000614 --DO 969	123600	85.295(1.6)	85.295(1.6)	0.001	4096	0.000(0)	0.132(0)	9.776(0)
000619--000622 -DO 331	200	0.099(0.0)	0.099(0.0)	0.000	4096	0.000(0)	0.000(0)	0.001(0)
000629--000635 -DO 7083	200	0.189(0.0)	0.002(0.0)	0.001	3	0.000(0)	0.000(0)	0.001(0)
000631--000633 --DO 7006	600	0.187(0.0)	0.187(0.0)	0.000	4096	0.000(0)	0.000(0)	0.042(0)
000639--000693 -DO 7000	200	4010.274(76.3)	5.819(0.1)	20.051	4096	0.000(0)	1.125(0)	2.614(0)
000641--000693 -DO 7000	819200	4004.455(76.1)	4004.455(76.1)	0.005	4096	0.000(0)	3.086(0)	198.186(4)
000698--000701 -DO 40	200	0.001(0.0)	0.001(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000699--000701 -DO 40	600	0.000(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000710--000721 -DO 47	200	2.548(0.0)	0.001(0.0)	0.013	2	0.000(0)	0.000(0)	0.000(0)
000713--000721 -DO 47	400	2.548(0.0)	0.004(0.0)	0.006	3	0.000(0)	0.000(0)	0.002(0)
000715--000719 --DO 41	1200	2.544(0.0)	2.544(0.0)	0.002	2048	0.000(0)	0.002(0)	0.707(0)
000727--000736 -DO 44	200	0.001(0.0)	0.001(0.0)	0.000	2	0.000(0)	0.000(0)	0.000(0)
000729--000731 -DO 42	400	0.000(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000732--000736 -DO 44	400	0.000(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000738--000741 -DO 43	200	0.000(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000743--000751 -DO 48	200	0.326(0.0)	0.001(0.0)	0.002	3	0.000(0)	0.000(0)	0.001(0)
000745--000749 -DO 45	600	0.325(0.0)	0.325(0.0)	0.001	4096	0.000(0)	0.000(0)	0.045(0)
000756--000758 -DO 46	200	0.000(0.0)	0.000(0.0)	0.000	2	0.000(0)	0.000(0)	0.000(0)
000761--000773 -DO 53	200	0.868(0.0)	0.002(0.0)	0.004	2	0.000(0)	0.000(0)	0.001(0)
000765--000771 -DO 51	400	0.866(0.0)	0.866(0.0)	0.002	2048	0.000(0)	0.001(0)	0.722(0)
000776--000779 -DO 52	200	0.001(0.0)	0.001(0.0)	0.000	2	0.000(0)	0.000(0)	0.000(0)
000788--000795 -DO 543	200	0.165(0.0)	0.002(0.0)	0.001	3	0.000(0)	0.000(0)	0.001(0)
000790--000793 -DO 541	600	0.163(0.0)	0.163(0.0)	0.000	2048	0.000(0)	0.000(0)	0.025(0)
000796--000803 -DO 544	200	0.164(0.0)	0.002(0.0)	0.001	3	0.000(0)	0.000(0)	0.001(0)
000798--000801 -DO 542	600	0.163(0.0)	0.163(0.0)	0.000	2048	0.000(0)	0.000(0)	0.025(0)
000812--000817 -DO 1012	200	1.071(0.0)	1.071(0.0)	0.005	4096	0.000(0)	0.001(0)	0.269(0)
000825--000827 -DO 1013	200	0.065(0.0)	0.065(0.0)	0.000	4096	0.000(0)	0.000(0)	0.007(0)
000844--000848 -DO 1023	200	0.000(0.0)	0.000(0.0)	0.000	0	0.000(0)	0.000(0)	0.000(0)
000851--000855 -DO 1024	200	0.058(0.0)	0.058(0.0)	0.000	2048	0.000(0)	0.000(0)	0.007(0)
000863--000867 -DO 5013	200	0.417(0.0)	0.417(0.0)	0.002	2048	0.000(0)	0.001(0)	0.358(0)
000874--000878 -DO 5014	200	0.417(0.0)	0.417(0.0)	0.002	2048	0.000(0)	0.000(0)	0.357(0)

000903-000912 -DO 33	200	0.232(0.0)	0.004(0.0)	0.001	3	0.000(0)	0.000(0)	0.001(0)
000905-000910 --DO 31	600	0.228(0.0)	0.228(0.0)	0.000	4096	0.000(0)	0.001(0)	0.046(0)
000923--000927 -DO 9003	1	0.000(0.0)	0.000(0.0)	0.000	3	0.000(0)	0.000(0)	0.000(0)
000924--000926 --DO 9004	3	0.000(0.0)	0.000(0.0)	0.000	4096	0.000(0)	0.000(0)	0.000(0)
000928--000934 -DO 9005	1	0.001(0.0)	0.000(0.0)	0.001	3	0.000(0)	0.000(0)	0.000(0)
000930--000932 --DO 9006	3	0.001(0.0)	0.001(0.0)	0.000	4096	0.000(0)	0.000(0)	0.000(0)

付録 C GYRO3D の ANALYZER-P による解析結果

(a) SUMMARY LIST

```

ANALYZER-P/SX REVISION : REV.260
ANALYZED DATE : Fri Mar 26 08:50:51 1999
FANP OPTIONS : -tm -amm -AL f=gyro3d2.anaout
                : lpunit all programs
                : -AO
                : do all programs
                : -ml

EXECUTION TIME : CPU TIME      = 0 : 02 ' 13 " 112 ( 133.112 sec)
                  ELAPSED TIME = 0 : 06 ' 21 " 763 ( 381.763 sec)

INSTRUCTION INFORMATION: INSTRUCTION COUNT      = 13334732023
                        FLOATING POINT ELEMENT COUNT = 2771380126
                        VECTOR INSTRUCTION COUNT = 442
                        VECTOR ELEMENT COUNT   = 5286

PERFORMANCE : 100.177 MOPS    20.820 MFLOPS (BY CPU TIME)

MEMORY INFORMATION : BANK CONFLICT TIME(sec) = 0.000( 0.0 %)
                      INST. CACHE MISS(sec) = 6.390( 4.8 %)
                      OPER. CACHE MISS(sec) = 44.759( 33.6 %)

VECTOR INFORMATION : VECTOR OPERATION RATIO = 0.00 %
                      AVERAGE VECTOR LENGTH = 12.0

```

(b) PROGRAM UNIT SUMMARY LIST

PROG.UNIT	ATR.	CODE	FREQUENCY	INCLUSIVE		EXCLUSIVE		MOPS	MFLOPS	V.OP.	AVER.	MEMORY LOSS TIME			
				CPU TIME(%)	CPU TIME(%)	RATIO	V.LEN					BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)	
FFT2B	SUB	600	36.677(27.6)	36.677(27.6)	102.3	13.6	0.00	0.0	0.000(0)	0.035(0)	11.833(9)				
PSH2	SUB	3993	61.098(45.9)	27.620(20.7)	102.6	4.4	0.00	0.0	0.000(0)	4.621(3)	9.166(7)				
USINM	SUB	300	10.689(8.0)	10.689(8.0)	46.6	21.1	0.00	0.0	0.000(0)	0.022(0)	6.355(5)				
		28744	10.307(7.7)	10.307(7.7)	163.9	64.1	0.00	0.0	0.000(0)	0.156(0)	1.430(1)				

PSH1	SUB	297	9.601(7.2)	9.601(7.2)	51.2	22.6	0.00	0.0	0.000(0)	0.020(0)	5.793(4)
HSINT	SUB	7186	21.687(16.3)	7.924(6.0)	126.9	24.9	0.00	0.0	0.000(0)	0.196(0)	1.252(1)
ATZFLD	SUB	199	38.400(28.8)	5.117(3.8)	98.8	18.2	0.00	0.0	0.000(0)	0.402(0)	1.380(1)
URFT	SUB	3993	4.820(3.6)	4.820(3.6)	108.9	53.1	0.00	0.0	0.000(0)	0.041(0)	1.131(1)
UPNR2	SUB	7986	3.829(2.9)	3.829(2.9)	102.2	0.0	0.00	0.0	0.000(0)	0.113(0)	0.945(1)
CFTN	SUB	3993	3.024(2.3)	3.024(2.3)	150.4	80.7	0.00	0.0	0.000(0)	0.066(0)	0.721(1)
MAIN_	MAIN	1	133.112(100.0)	2.052(1.5)	111.1	29.2	0.00	25.2	0.000(0)	0.152(0)	0.501(0)
BOUND	SUB	600	2.035(1.5)	2.035(1.5)	16.9	4.8	0.00	0.0	0.000(0)	0.002(0)	1.427(1)
MIRROR	SUB	1400	1.967(1.5)	1.967(1.5)	57.4	8.7	0.00	0.0	0.000(0)	0.078(0)	1.191(1)
PEEK	SUB	100	1.511(1.1)	1.499(1.1)	101.2	11.6	0.00	0.0	0.000(0)	0.135(0)	0.343(0)
PNR	SUB	3993	1.299(1.0)	1.299(1.0)	81.5	0.1	0.00	0.0	0.000(0)	0.029(0)	0.384(0)
UCOSM	SUB	3200	1.110(0.8)	1.110(0.8)	166.4	66.2	0.00	0.0	0.000(0)	0.020(0)	0.168(0)
FCOST	SUB	800	2.556(1.9)	1.060(0.8)	132.8	26.3	0.00	0.0	0.000(0)	0.019(0)	0.180(0)
EBFORM	SUB	200	0.915(0.7)	0.915(0.7)	107.1	37.4	0.00	0.0	0.000(0)	0.072(0)	0.234(0)
EXTEND	SUB	1000	0.693(0.5)	0.693(0.5)	105.4	3.0	0.00	0.0	0.000(0)	0.111(0)	0.191(0)
ENERGY	SUB	10	0.317(0.2)	0.317(0.2)	99.9	35.1	0.00	0.0	0.000(0)	0.011(0)	0.013(0)
RFT	SUB	3993	9.235(6.9)	0.092(0.1)	8.3	0.0	0.00	0.0	0.000(0)	0.035(0)	0.046(0)
PSH0	SUB	3	0.085(0.1)	0.085(0.1)	57.9	25.6	0.00	0.0	0.000(0)	0.000(0)	0.045(0)
REVERS	FUNC	32768	0.081(0.1)	0.081(0.1)	58.2	11.8	0.00	0.0	0.000(0)	0.013(0)	0.001(0)
PLOT1	SUB	219	0.056(0.0)	0.056(0.0)	51.3	0.8	0.06	5.9	0.000(0)	0.012(0)	0.001(0)
PLOTN	ENT	39									
PLOT2	ENT	39									
PLOT3	ENT	63									
PLOT4	ENT	19									
PLOT4R	ENT	20									
INTSTC	SUB	4	0.158(0.1)	0.043(0.0)	63.2	19.1	0.00	0.0	0.000(0)	0.002(0)	0.009(0)
VELSET	SUB	8192	0.070(0.1)	0.033(0.0)	54.2	18.3	0.00	0.0	0.000(0)	0.007(0)	0.001(0)
ELDAT	SUB	2	0.030(0.0)	0.030(0.0)	91.1	2.2	0.01	68.0	0.000(0)	0.001(0)	0.001(0)
POTOTC	SUB	3	0.019(0.0)	0.018(0.0)	61.4	1.3	0.02	68.0	0.000(0)	0.004(0)	0.000(0)
POTOTE	SUB	3	0.017(0.0)	0.017(0.0)	65.1	1.4	0.02	68.0	0.000(0)	0.003(0)	0.000(0)
POTOUT	SUB	3	0.017(0.0)	0.016(0.0)	63.5	1.3	0.03	68.0	0.000(0)	0.003(0)	0.000(0)
ESTIME	SUB	4662	0.013(0.0)	0.013(0.0)	30.1	0.4	0.00	0.0	0.000(0)	0.004(0)	0.003(0)
UTABT	SUB	7986	0.013(0.0)	0.013(0.0)	34.4	0.0	0.00	0.0	0.000(0)	0.001(0)	0.006(0)
COUNTP	SUB	10	0.009(0.0)	0.009(0.0)	56.7	9.4	0.00	0.0	0.000(0)	0.000(0)	0.002(0)
BXYDAT	SUB	3	0.007(0.0)	0.007(0.0)	101.7	0.0	0.04	68.0	0.000(0)	0.000(0)	0.002(0)

(c) PROGRAM UNIT DETAIL LIST (SRC)

LINE	NESTING	FREQUENCY	INCLUSIVE		AVERAGE P/V	AVERAGE	MEMORY LOSS TIME			
			CPU TIME(%)	CPU TIME(%)			LOOP.LEN.	BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)
002397	SRC	600	36.677(27.6)	0.004(0.0)	0.061		0.000(0)	0.002(0)	0.001(0)	
002428--002431 -DO 10		400	4.545(3.4)	4.545(3.4)	0.011	473344	0.000(0)	0.002(0)	0.001(0)	
002441--002560 -DO 20		400	10.106(7.6)	0.018(0.0)	0.025	8	0.000(0)	0.004(0)	0.007(0)	
002448--002529 --DO 30		3200	3.081(2.3)	3.081(2.3)	0.001	512	0.000(0)	0.005(0)	1.215(1)	
002531--002541 --DO 90		3200	0.459(0.3)	0.459(0.3)	0.000	512	0.000(0)	0.003(0)	0.106(0)	
002547--002556 --DO 40		3200	6.548(4.9)	0.105(0.1)	0.002	32	0.000(0)	0.002(0)	0.065(0)	
002550--002555 --DO 50		102400	6.443(4.8)	6.443(4.8)	0.000	128	0.000(0)	0.004(0)	4.049(3)	
002565--002570 -DO 60		400	22.022(16.5)	0.037(0.0)	0.055	16	0.000(0)	0.004(0)	0.019(0)	
002566--002569 --DO 70		6400	21.984(16.5)	21.984(16.5)	0.003	29584	0.000(0)	0.010(0)	6.369(5)	

(d) PROGRAM UNIT DETAIL LIST (FFT2B)

LINE	NESTING	FREQUENCY	INCLUSIVE		AVERAGE P/V	AVERAGE	MEMORY LOSS TIME			
			CPU TIME(%)	CPU TIME(%)			LOOP.LEN.	BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)
004585	FFT2B	3993	61.098(45.9)	0.135(0.1)	0.015		0.000(0)	0.054(0)	0.053(0)	
004634--004638 -DO 110		1495	2.959(2.2)	0.018(0.0)	0.002	33	0.000(0)	0.001(0)	0.019(0)	
004635--004638 --DO 110		49335	2.942(2.2)	0.129(0.1)	0.000	16	0.000(0)	0.000(0)	0.059(0)	
004636--004638 --DO 110		789360	2.812(2.1)	2.812(2.1)	0.000	33	0.000(0)	0.001(0)	2.072(2)	
004642	-FSINT	1495	4.509(3.4)	1.646(1.2)	0.003		0.000(0)	0.040(0)	0.259(0)	
004644	-FCOST	0	0.000(0.0)	0.000(0.0)	0.000		0.000(0)	0.000(0)	0.000(0)	
004649--004653 -DO 120		1495	1.421(1.1)	0.006(0.0)	0.001	33	0.000(0)	0.001(0)	0.000(0)	
004650--004653 --DO 120		49335	1.415(1.1)	0.190(0.1)	0.000	16	0.000(0)	0.016(0)	0.000(0)	
004651--004653 ---DO 120		789360	1.225(0.9)	1.225(0.9)	0.000	33	0.000(0)	0.133(0)	0.296(0)	
004671--004675 -DO 210		1495	0.937(0.7)	0.015(0.0)	0.001	33	0.000(0)	0.003(0)	0.002(0)	
004672--004675 --DO 210		49335	0.922(0.7)	0.214(0.2)	0.000	33	0.000(0)	0.004(0)	0.058(0)	
004673--004675 --DO 210		1628055	0.707(0.5)	0.707(0.5)	0.000	16	0.000(0)	0.012(0)	0.133(0)	
004679	-FSINT	1495	4.507(3.4)	1.648(1.2)	0.003		0.000(0)	0.040(0)	0.261(0)	
004681	-FCOST	0	0.000(0.0)	0.000(0.0)	0.000		0.000(0)	0.000(0)	0.000(0)	
004686--004690 -DO 220		1495	2.521(1.9)	0.026(0.0)	0.002	33	0.000(0)	0.015(0)	0.000(0)	
004687--004690 --DO 220		49335	2.495(1.9)	1.227(0.9)	0.000	33	0.000(0)	0.588(0)	0.001(0)	
004688--004690 ---DO 220		1628055	1.268(1.0)	1.268(1.0)	0.000	16	0.000(0)	0.496(0)	0.276(0)	
004708--004712 -DO 310		1495	1.254(0.9)	0.001(0.0)	0.001	16	0.000(0)	0.001(0)	0.000(0)	
004709--004712 --DO 310		23920	1.253(0.9)	0.064(0.0)	0.000	33	0.000(0)	0.000(0)	0.001(0)	

004710--004712 --DO 310	789360	1.188(0.9)	1.188(0.9)	0.000	33	0.000(0)	0.000(0)	0.348(0)
004715 -RFT	1495	3.496(2.6)	0.037(0.0)	0.002		0.000(0)	0.014(0)	0.019(0)
004719--004724 -DO 320	1495	0.963(0.7)	0.006(0.0)	0.001	16	0.000(0)	0.004(0)	0.000(0)
004721--004724 --DO 320	23920	0.957(0.7)	0.065(0.0)	0.000	33	0.000(0)	0.000(0)	0.001(0)
004722--004724 --DO 320	789360	0.892(0.7)	0.892(0.7)	0.000	33	0.000(0)	0.000(0)	0.187(0)
004747--004752 -DO 410	2498	2.071(1.6)	0.008(0.0)	0.001	16	0.000(0)	0.003(0)	0.000(0)
004749--004752 --DO 410	39968	2.063(1.5)	0.123(0.1)	0.000	33	0.000(0)	0.000(0)	0.003(0)
004750--004752 --DO 410	1318944	1.940(1.5)	1.940(1.5)	0.000	33	0.000(0)	0.001(0)	0.618(0)
004755 -RFT	2498	5.739(4.3)	0.055(0.0)	0.002		0.000(0)	0.021(0)	0.027(0)
004758--004762 -DO 420	2498	2.158(1.6)	0.008(0.0)	0.001	16	0.000(0)	0.001(0)	0.000(0)
004759--004762 --DO 420	39968	2.150(1.6)	0.425(0.3)	0.000	33	0.000(0)	0.091(0)	0.001(0)
004760--004762 --DO 420	1318944	1.725(1.3)	1.725(1.3)	0.000	33	0.000(0)	0.506(0)	0.375(0)
004773--004777 -DO 510	2498	3.581(2.6)	0.010(0.0)	0.001	33	0.000(0)	0.001(0)	0.003(0)
004774--004777 --DO 510	82434	3.491(2.6)	1.109(0.8)	0.000	33	0.000(0)	0.014(0)	0.100(0)
004775--004777 --DO 510	2720322	2.382(1.8)	2.382(1.8)	0.000	16	0.000(0)	1.275(1)	0.227(0)
004781 -FSINT	2098	6.339(4.8)	2.314(1.7)	0.003		0.000(0)	0.058(0)	0.365(0)
004783 -FCOST	400	1.277(1.0)	0.530(0.4)	0.003		0.000(0)	0.009(0)	0.090(0)
004787--004791 -DO 520	2498	2.326(1.7)	0.036(0.0)	0.001	33	0.000(0)	0.011(0)	0.001(0)
004788--004791 --DO 520	82434	2.289(1.7)	0.967(0.7)	0.000	33	0.000(0)	0.436(0)	0.004(0)
004789--004791 --DO 520	2720322	1.322(1.0)	1.322(1.0)	0.000	16	0.000(0)	0.308(0)	0.267(0)
004801--004805 -DO 610	2498	5.989(4.5)	0.095(0.1)	0.002	33	0.000(0)	0.037(0)	0.046(0)
004802--004805 --DO 610	82434	5.894(4.4)	0.875(0.7)	0.000	16	0.000(0)	0.329(0)	0.099(0)
004803--004805 --DO 610	1318944	5.019(3.8)	5.019(3.8)	0.000	33	0.000(0)	0.275(0)	3.491(3)
004809 -FSINT	2098	6.332(4.8)	2.316(1.7)	0.003		0.000(0)	0.057(0)	0.367(0)
004811 -FCOST	400	1.279(1.0)	0.530(0.4)	0.003		0.000(0)	0.010(0)	0.090(0)
004815--004819 -DO 620	2498	1.385(1.0)	0.005(0.0)	0.001	33	0.000(0)	0.003(0)	0.000(0)
004816--004819 --DO 620	82434	1.380(1.0)	0.096(0.1)	0.000	16	0.000(0)	0.000(0)	0.002(0)
004817--004819 --DO 620	1318944	1.285(1.0)	1.285(1.0)	0.000	33	0.000(0)	0.001(0)	0.422(0)

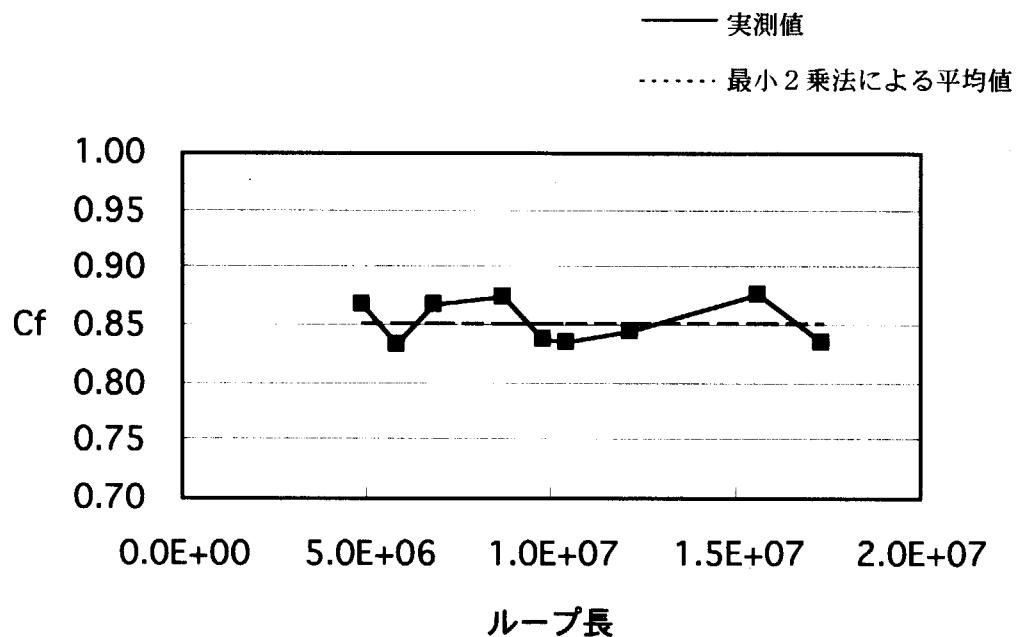
(e) PROGRAM UNIT DETAIL LIST (PSH2)

LINE	NESTING	FREQUENCY	INCLUSIVE		EXCLUSIVE		AVERAGE P/V	AVERAGE	MEMORY LOSS TIME		
			CPU TIME(%)	CPU TIME(%)	CPU TIME	LOOP,LEN.			BANK CONF.(%)	INST,CACHE(%)	OPER,CACHE(%)
002984	PSH2	300	10.689(8.0)	0.002(0.0)	0.036				0.000(0)	0.001(0)	0.000(0)
003042--003184 -DO 10		200	10.687(8.0)	0.012(0.0)	0.053	8			0.000(0)	0.003(0)	0.004(0)
003047--003182 -DO 20		1500	10.675(8.0)	10.675(8.0)	0.007	512			0.000(0)	0.018(0)	6.351(5)

(f) PROGRAM UNIT DETAIL LIST (PSH1)

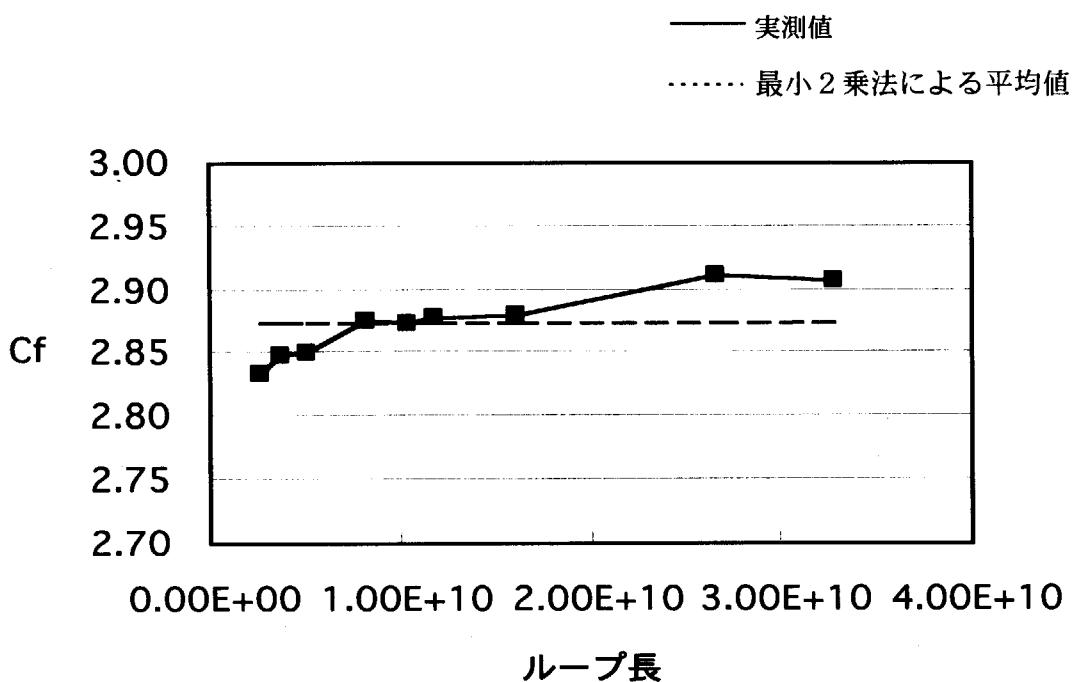
LINE	NESTING	FREQUENCY	INCLUSIVE		EXCLUSIVE		P/V	AVERAGE	MEMORY LOSS TIME			
			CPU TIME(%)	CPU TIME(%)	CPU TIME	LOOP LEN.			BANK CONF.(%)	INST.CACHE(%)	OPER.CACHE(%)	
002781	PSH1	297	9.601(7.2)	0.002(0.0)	0.032		0.000(0)	0.001(0)	0.000(0)			
002838-002980 -00 10		198	9.599(7.2)	0.012(0.0)	0.048	8	0.000(0)	0.003(0)	0.004(0)			
002843-002978 -00 20		1584	9.587(7.2)	9.587(7.2)	0.006	512	0.000(0)	0.016(0)	5.789(4)			

付録 D キャッシュミス影響予測子 Cf と計測結果



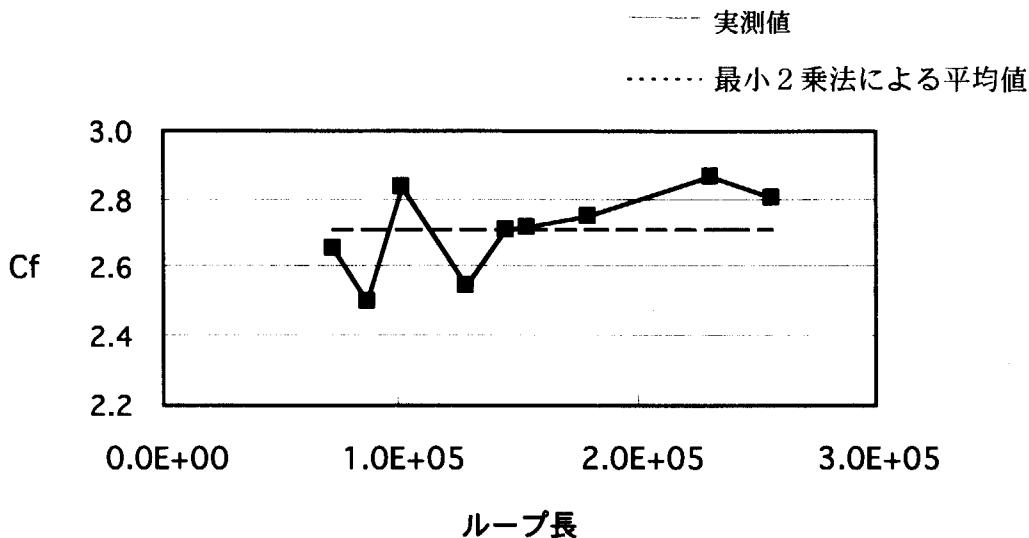
このループの場合、データの先読み機構(prefetch 命令)やキャッシュ上のデータの再利用性が高い。測定値は理論性能値と一致している。

図1 force ルーチン



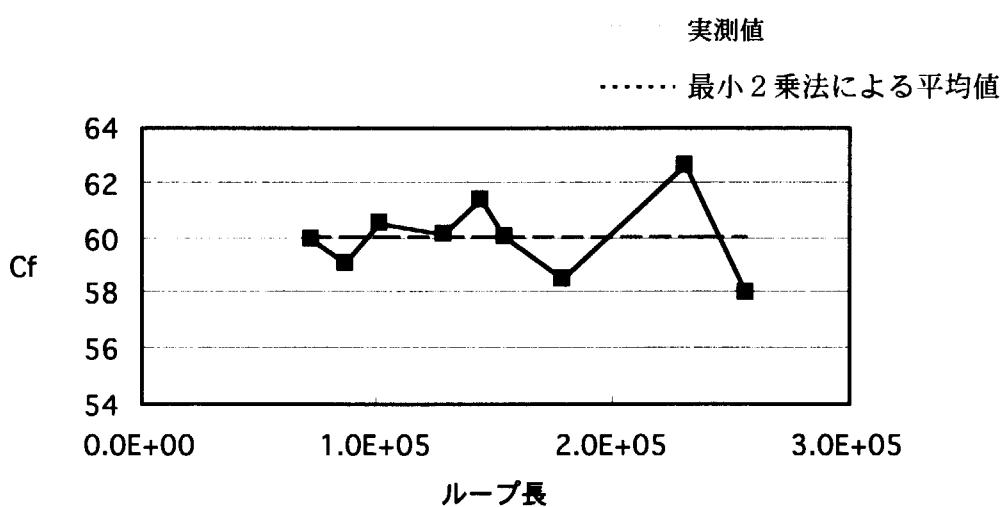
参照アクセスする変数データ量が大きい。データの先読み機構(prefetch 命令)は効いているが、データの再利用性は図1の force と比べると低く(再利用性は、約半分)、実行時間は理論値(ヒット率 100%の場合)の3倍程度となっている。

図2 table ルーチン



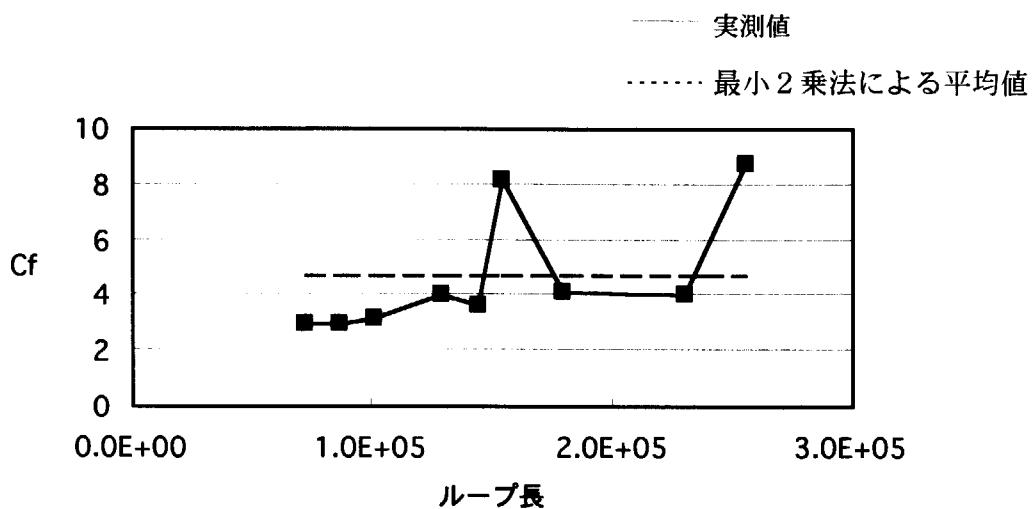
粒子数の計算ループのため、データ規模は粒子数に依存する。データ規模は大きいが、データの再利用性は、図2のtableと同様である。
そのため、図2に同様の傾向を示している。

図3 Do309 - Do305 Propcel ルーチン



ループ内に入出力文を含んでいるので、一部キャッシュミスが発生している。(キャッシュミス多発時には、理論値の1オーダー遅いという現象が発生している。) また、計算処理以外の入出力処理時間も計測値として評価されているため、Cfが60と大きくなっている。

図4 Other time of Propcel ルーチン



ループの特徴としては、データの再利用性がほとんどない。データの先読み機構 (prefetch 命令)のみが効いており、理論性能値の約 5 倍の実行時間となっている。

図 5 pmove ルーチン

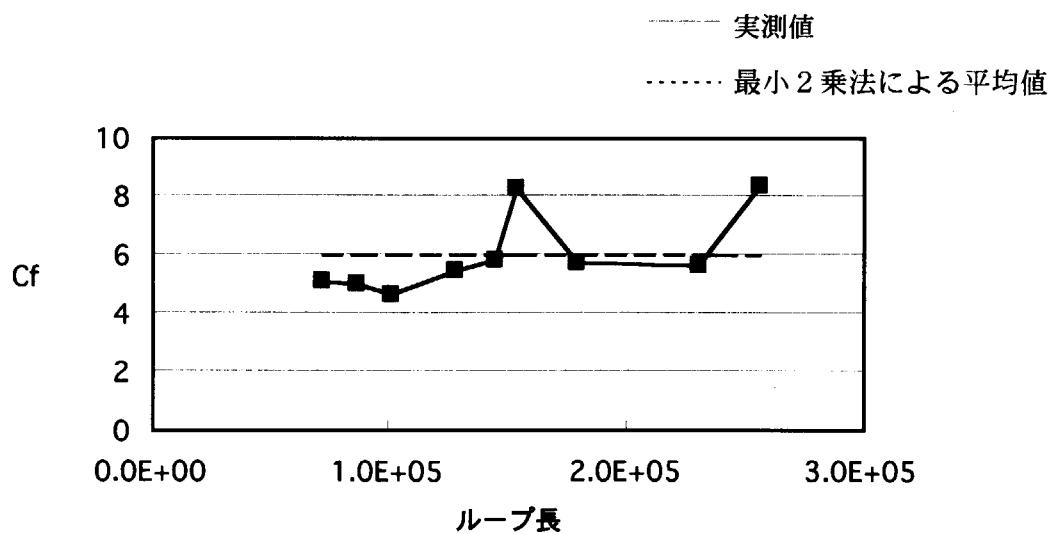
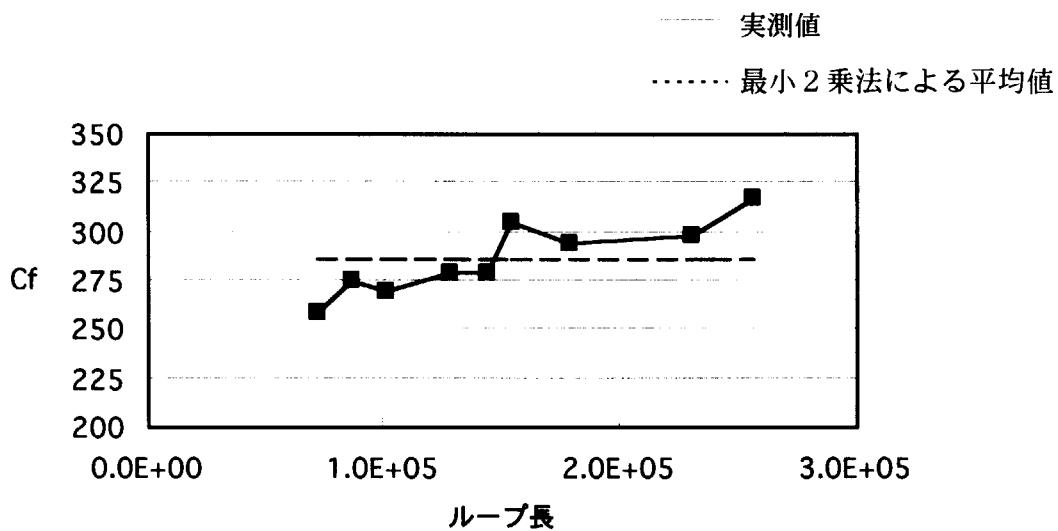


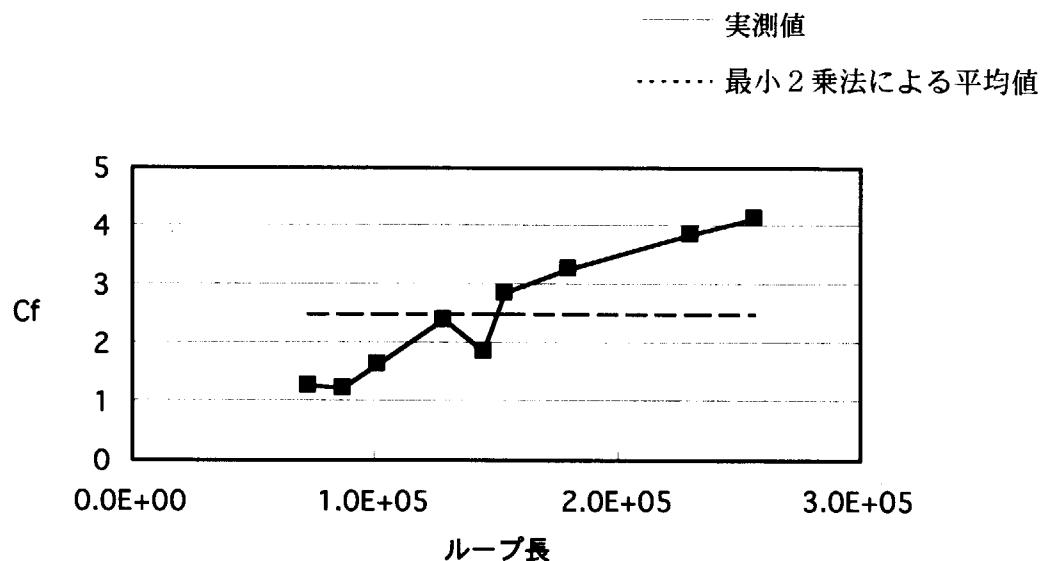
図 5 の pmove ルーチンと同様のループ構造となっている。

図 6 pcross ルーチン



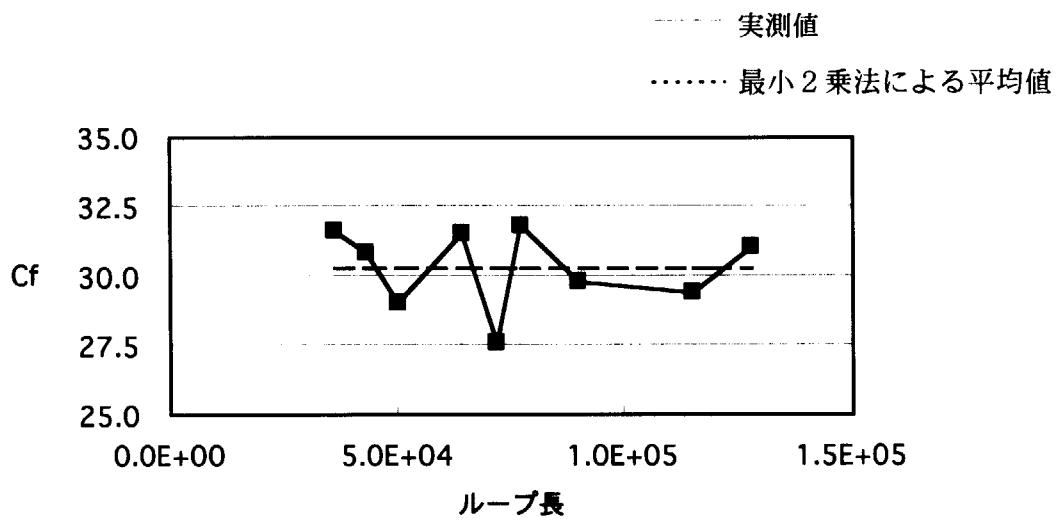
計測時間の大部分が入出力文であるため、ほとんどメモリ上のデータアクセスとなっている。データのキャッシュミス現象として C_f が 300 程度と極めて大きくなっている。(ヒット率 100% の場合と比較して 200~300 倍遅くなるという現象が現れている。)

図7 Outside of 224 Loop, main ルーチン



ロード・ストアのコストと演算コストが同程度となっている。演算数は、ループ長に応じて決まるため、 C_f の値が増大する傾向であるが、ここではロード・ストアが性能を決める要因と考え定数化を試みた。
 C_f を理論性能値の 3 倍程度とした。

図 8 Inside of 224 Loop, main ルーチン



キャッシュのヒット率が低い。キャッシュミス多発時には、理論値の1オーダー遅いという現象を表している。図3 Other time of Propcel ルーチンでも同様の傾向であったが、キャッシュミス多発時の代表的な例である。

図9 maxwell ルーチン

付録 E 性能シミュレータの使用方法

(1) 解析パラメタによる実行時間の予測

ここでは、MD0 コードの解析条件パラメタを指定する。

<解析パラメタによる実行時間の予測>		
(1) 解析条件パラメタ		
a. 粒子数密度	= 0.3	←粒子数密度を入力する.
b. アスペクト比	= 2.016	
c. RINT	= 4	←RINT 変数を入力する.
d. Cut Off 距離	= 3	←Cut Off 距離を入力する.
e. 上面境界温度	= 1200	←上面境界温度を入力する.
f. 下面境界温度	= 120	←下面境界温度を入力する.
g. 解析時間ステップ	= 0.01	←解析時間ステップを入力する.
σ : 原子の直径が 1		
(2) サンプリングセルパラメタ		
a. X 方向セル数	= 80	←X 方向セル数を入力する.
b. Y 方向セル数	= 100	←Y 方向セル数を入力する.
c. XY 方向衝突セル	= 3	←XY 方向衝突セル数を入力する.
総衝突セル数	= 72000	個
(3) 時間ステップパラメタ		
a. Isnap	= 12	←Isnap 変数を入力する.
b. Itimax	= 20	←Itimax 変数を入力する.
c. 評価ステップ間隔	= 20	←評価ステップ間隔を入力する.

(2) ハードウェアによる加速係数の指定

ここではハードウェアによる加速により、MD0 コードの実行時間がどの程度短縮されるかを検討する場合に、ルーチン毎の短縮効果を指定する。

B.<ハードウェアによる加速>

(1)ルーチン毎のコストの短縮

ルーチン名	L/S コスト	演算コスト	短縮係数	実行時間	演算時間	時間比率
force	1.325E+08	9.723E+07	1.00	① 40.84	35.13	0.43
maxwell	7.200E+05	1.203E+07	1.00	② 0.03	0.02	0.00
propcel1	2.184E+06	1.988E+06	1.00	2.13	0.72	0.02
propcel2	8.160E+05	1.480E+06	1.00	③ 0.28	0.03	0.01
pmoves	1.296E+06	7.972E+05	1.00	2.19	0.29	0.02
pcross	1.296E+06	1.485E+06	1.00	2.80	0.53	0.03
table	1.037E+10	1.709E+10	# 1.00	④ 44.70	25.62	0.47
main1	4.320E+05	2.882E+05	1.00	⑤ 0.19	0.00	0.00
main2	1.730E+06	7.260E+05	1.00	⑥ 1.56	0.26	0.02
総和：	1.051E+10	1.720E+10		95.31	62.59	1.00

(2)CPU の動作周波数の設定

667 MHz ④

①force ルーチンの短縮係数

force ルーチンの実行時間は全体の計算時間に占める割合が大きいため，“ルーチン毎の基本コスト”シートで時間短縮効果を詳細に設定する。“ルーチン毎の基本コスト”シートで計算された短縮係数がこのセルに反映される。

②ルーチン毎の短縮係数

これらのルーチンの実行時間は全体の計算時間に占める割合が小さいため、直接、短縮係数を指定する。シミュレータは予測時間に指定された短縮係数を乗じた時間を新たに予測時間とする。

③table ルーチンの短縮係数

table ルーチンも実行時間が全体の計算時間に占める割合が大きいため，“ルーチン毎の基本コスト”シートで時間短縮効果を詳細に設定する。“ルーチン毎の基本コスト”シートで計算された短縮係数がこのセルに反映される。

④CPU の動作周波数

CPU の動作周波数を指定する。シミュレータが使用する基本コストは Alpha 21264 667MHz のものであり、ここで CPU の動作周波数を指定すると、シミュレータは指定された動作周波数と 667MHz の比から基本コストを修正して予測時間を出力する。

(3) 結果表示

ここではシミュレータによる予測結果が表示される。

予測実行時間	= 95.31	(秒)	①	加速後予測時間	95.31	(秒)	③
使用メモリサイズ	= 115.78	Mb	②	実行短縮時間	0.00	(秒)	④
				実行時間短縮率	1.00	倍	⑤

①予測実行時間

コードの予測実行時間が表示される。

②使用メモリサイズ

コードが使用するメモリサイズが表示される。

③加速後予測時間

ハードウェアによる加速を指定した場合、加速後の予測時間が表示される。

④実行短縮時間

ハードウェアによる加速を指定した場合、短縮された時間が表示される。

⑤実行時間短縮率

ハードウェアによる加速を指定した場合、加速前と加速後の予測時間の比が表示される。

ここでは force ルーチンと table ルーチンについて、処理毎に専用ハードウェア等による計算時間の短縮効果を指定する。

ルーチン毎の基本コスト（コスト単位：実数加算演算を 1 とした時の値）

(A) force ルーチン				
処理内容	DO ループ 1 回当たりの ロードストア 演算 コスト コスト			
相互力の計算	8	①	21.85	②
粒子間距離の計算	13	③	8.59	④
座標値の取得	5	⑤	0	⑥
変数のクリア	4	⑦	0	⑧
	30		30.44	
総和				
force ルーチン短縮時間	=	0.00	(秒)	⑨
(B) table ルーチン				
インデックスの代入	1	⑩	1	⑪
粒子間距離の計算	4	⑫	6.59	⑬
座標値の取得	6	⑭	2.07	⑮
	11		9.66	
総和				
table ルーチン短縮時間	=	0.00	(秒)	⑯
加速後予測時間	=	95.31	(秒)	⑰
実行短縮時間	=	0.00	(秒)	⑱
実行時間短縮率	=	1.00	倍	⑲

①～⑧は force ルーチンに関する処理毎のロード／ストア、及び演算コストである。専用ハードウェア等によりコストを短縮する場合は、セルの数字を書き直す。

- ① 相互力の計算に関するロード／ストアコスト
- ② 相互力の計算に関する演算コスト
- ③ 粒子間距離の計算に関するロード／ストアコスト
- ④ 粒子間距離の計算に関する演算コスト
- ⑤ 座標値の取得に関するロード／ストアコスト
- ⑥ 座標値の取得に関する演算コスト

⑦変数のクリアに関するロード／ストアコスト

⑧変数のクリアに関する演算コスト

⑨は①～⑧で指定された値に応じて, force ルーチンの短縮時間が計算され,
表示される.

⑩～⑯は table ルーチンに関する処理毎のロード／ストア, 及び演算コスト
である. 専用ハードウェア等によりコストを短縮する場合は, セルの数字を
書き直す.

⑩インデックスの代入に関するロード／ストアコスト

⑪インデックスの代入に関する演算コスト

⑫粒子間距離の計算に関するロード／ストアコスト

⑬粒子間距離の計算に関する演算コスト

⑭座標値の取得に関するロード／ストアコスト

⑮座標値の取得に関する演算コスト

⑯は⑩～⑯で指定された値に応じて, force ルーチンの短縮時間が計算され,
表示される.

⑰は md0 コードの加速後の予測時間が表示される.

⑱は md0 コードの加速による実行短縮時間が表示される.

⑲は md0 コードの加速による実行時間短縮率が表示される.

国際単位系(SI)と換算表

表1 SI基本単位および補助単位

量	名称	記号
長さ	メートル	m
質量	キログラム	kg
時間	秒	s
電流	アンペア	A
熱力学温度	ケルビン	K
物質量	モル	mol
光度	カンデラ	cd
平面角	ラジアン	rad
立体角	ステラジアン	sr

表3 固有の名称をもつSI組立単位

量	名称	記号	他のSI単位による表現
周波数	ヘルツ	Hz	s^{-1}
力	ニュートン	N	$m \cdot kg/s^2$
圧力、応力	パスカル	Pa	N/m^2
エネルギー、仕事、熱量	ジュール	J	$N \cdot m$
功率、放射束	ワット	W	J/s
電気量、電荷	クーロン	C	$A \cdot s$
電位、電圧、起電力	ボルト	V	W/A
静電容量	ファラード	F	C/V
電気抵抗	オーム	Ω	V/A
コンダクタンス	ジーメンス	S	A/V
磁束密度	ウェーバー	Wb	$V \cdot s$
磁束密度	テスラ	T	Wb/m^2
インダクタンス	ヘンリイ	H	Wb/A
セルシウス温度	セルシウス度	°C	
光束度	ルーメン	lm	$cd \cdot sr$
照度	ルクス	lx	lm/m^2
放射能	ベクレル	Bq	s^{-1}
吸収線量	グレイ	Gy	J/kg
線量当量	シーベルト	Sv	J/kg

表2 SIと併用される単位

名称	記号
分、時、日	min, h, d
度、分、秒	°, ', "
リットル	l, L
トン	t
電子ボルト	eV
原子質量単位	u

$$1 \text{ eV} = 1.60218 \times 10^{-19} \text{ J}$$

$$1 \text{ u} = 1.66054 \times 10^{-27} \text{ kg}$$

表5 SI接頭語

倍数	接頭語	記号
10^{18}	エクサ	E
10^{15}	ペタ	P
10^{12}	テラ	T
10^9	ギガ	G
10^6	メガ	M
10^3	キロ	k
10^2	ヘクト	h
10^1	デカ	da
10^{-1}	デシ	d
10^{-2}	センチ	c
10^{-3}	ミリ	m
10^{-6}	マイクロ	μ
10^{-9}	ナノ	n
10^{-12}	ピコ	p
10^{-15}	フェムト	f
10^{-18}	アト	a

(注)

- 表1—5は「国際単位系」第5版、国際度量衡局1985年刊行による。ただし、1eVおよび1uの値はCODATAの1986年推奨値によった。
- 表4には海里、ノット、アール、ヘクタールも含まれているが日常の単位なのでここでは省略した。
- barは、JISでは流体の圧力を表わす場合に限り表2のカテゴリーに分類されている。
- EC閣僚理事会指令ではbar、barnおよび「血圧の単位」mmHgを表2のカテゴリーに入れている。

換算表

力	N($=10^5 \text{ dyn}$)	kgf	lbf
	1	0.101972	0.224809
	9.80665	1	2.20462
	4.44822	0.453592	1

粘度 $1 \text{ Pa} \cdot \text{s} (\text{N} \cdot \text{s}/\text{m}^2) = 10 \text{ P} (\text{ポアズ}) (\text{g}/(\text{cm} \cdot \text{s}))$

動粘度 $1 \text{ m}^2/\text{s} = 10^4 \text{ St} (\text{ストーカス}) (\text{cm}^2/\text{s})$

圧力	MPa($=10 \text{ bar}$)	kgf/cm ²	atm	mmHg(Torr)	lbf/in ² (psi)
力	1	10.1972	9.86923	7.50062×10^3	145.038
	0.0980665	1	0.967841	735.559	14.2233
	0.101325	1.03323	1	760	14.6959
	1.33322×10^{-4}	1.35951×10^{-3}	1.31579×10^{-3}	1	1.93368×10^{-2}
	6.89476×10^{-3}	7.03070×10^{-2}	6.80460×10^{-2}	51.7149	1

エネルギー・仕事・熱量	J($=10^7 \text{ erg}$)	kgf·m	kW·h	cal(計量法)	Btu	ft · lbf	eV	1 cal = 4.18605 J(計量法)
	1	0.101972	2.77778×10^{-7}	0.238889	9.47813×10^{-4}	0.737562	6.24150×10^{-18}	$= 4.184 \text{ J} (\text{熱化学})$
	9.80665	1	2.72407×10^{-6}	2.34270	9.29487×10^{-3}	7.23301	6.12082×10^{-19}	$= 4.1855 \text{ J} (15^\circ \text{C})$
	3.6×10^6	3.67098×10^5	1	8.59999×10^5	3412.13	2.65522×10^6	2.24694×10^{25}	$= 4.1868 \text{ J} (\text{国際蒸気表})$
	4.18605	0.426858	1.16279×10^{-6}	1	3.96759×10^{-3}	3.08747	2.61272×10^{-19}	仕事率 1 PS(仮馬力)
	1055.06	107.586	2.93072×10^{-4}	252.042	1	778.172	6.58515×10^{21}	$= 75 \text{ kgf} \cdot \text{m/s}$
	1.35582	0.138255	3.76616×10^{-7}	0.323890	1.28506×10^{-3}	1	8.46233×10^{18}	$= 735.499 \text{ W}$
	1.60218×10^{-19}	1.63377×10^{-20}	4.45050×10^{-28}	3.82743×10^{-20}	1.51857×10^{-22}	1.18171×10^{-19}	1	

放射能	Bq	Ci
	1	2.70270×10^{-11}
	3.7×10^{10}	1

吸収線量	Gy	rad
	1	100
	0.01	1

照射線量	C/kg	R
	1	3876
	2.58×10^{-4}	1

線量当量	Sv	rem
	1	100
	0.01	1

(86年12月26日現在)

次世代計算機概念検討－平成11年度作業報告－