

JAERI-Data/Code
94-021



原子力コードのベクトル化と改良 (III)
[DGR, STREAM V3.1, Cella, GGR]

1995年1月

根本俊行*・江口則地*・渡辺秀雄*
町田昌彦*・横川三津夫・藤井実

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、日本原子力研究所技術情報部情報資料課（〒319-11 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（〒319-11 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 Information Division, Department of Technical Information, Japan Atomic Energy Research Institute, Tokaimura, Naka-gun, Ibaraki-ken 319-11, Japan.

© Japan Atomic Energy Research Institute, 1994

編集兼発行 日本原子力研究所

印 刷 株原子力資料サービス

原子力コードのベクトル化と改良（Ⅲ）
[DGR, STREAM V3.1, Cell, GGR]

日本原子力研究所東海研究所情報システムセンター
根本 俊行*・江口 則地*・渡辺 秀雄*・町田 昌彦*
横川三津夫・藤井 実

(1994年12月6日受理)

日本原子力研究所・情報システムセンターは、平成5年度に原子力コードのベクトル化及び高速化のための改良を行った。ベクトル化及び高速化を行ったコードはダイヤモンド型結晶の放射線照射損傷をシミュレーションする分子動力学コードDGR, 3次元非定常圧縮性流体用解析コードSTREAM V3.1, セルオートマトンモデルを用いた2次元流体シミュレーションコードCella, 及び黒鉛型結晶における放射線照射損傷をシミュレーションする分子動力学コードGGRである。ベクトル化による速度性能向上はVP2600でのスカラー実行に比較してDGRで8.2倍, STREAM V3.1で6.8～14.8倍, Cellaで15～16倍, そしてGGRでは1.23倍を得た。本報告書では、上記原子力コードのベクトル化手法, ベクトル化効果, 計算結果の評価及びコードの改良方法などについて記す。

Vectorization and Improvement of Nuclear Codes (III)

[DGR, STREAM V3.1, Cell, GGR]

Toshiyuki NEMOTO*, Norikuni EGUCHI*, Hideo WATANABE*
Masahiko MACHIDA*, Mitsuo YOKOKAWA and Minoru FUJII

Computing and Information Systems Center
Tokai Research Establishment
Japan Atomic Energy Research Institute
Tokai-mura, Naka-gun, Ibaraki-ken

(Received December 6, 1994)

Four nuclear codes have been vectorized and improved in order to realize the high speed performance on the VP2600 supercomputer at Computing and Information Systems Center of JAERI in the fiscal year 1993.

Molecular Dynamics simulation code DGR which simulates the irradiation damage on diamond crystalline, three-dimensional non-steady compressible fluid dynamics code STREAM V3.1, two-dimensional fluid simulation code using Cell Automaton model Cell and Molecular Dynamics code GGR which simulates the irradiation damage on black carbon crystalline have been vectorized and improved, respectively.

Speed up ratios by the vectorization to scalar mode on VP2600 show 2.8, 6.8 ~ 14.8, 15 ~ 16 and 1.23 times for DGR, STREAM V3.1, Cell and GGR, respectively.

In this report, we present vectorization techniques, vectorization effects, evaluations of the numerical results and techniques for the improvement.

Keywords : Nuclear Codes, Vectorization, Molecular Dynamics, DGR, GGR, STREAM V.3.1, Cellular Automata, Cell

* On leave from FUJITSU, Ltd

目 次

1. はじめに	1
2. DGR のベクトル化	2
2.1 コード概要	2
2.2 FORTRAN77EX への対応	2
2.3 ベクトル化	4
2.4 ベクトル化の効果	7
2.5 実行環境	8
3. STREAM V3.1 のベクトル化	49
3.1 基礎方程式	49
3.2 ベクトル化方法	52
3.3 ベクトル化の結果	54
4. CELLA のベクトル化	58
4.1 動的解析	58
4.2 ベクトル化方法	59
4.3 計算結果の評価	61
4.4 ベクトル化効果	61
4.5 まとめ	62
5. GGR の高速化	77
5.1 コード概要	77
5.2 FORTRAN77EX への対応	77
5.3 ベクトル化	78
5.4 ベクトル化の効果	81
5.5 実行環境	82
5.6 まとめ	82
6. おわりに	118
謝 辞	118
参考文献	119

Contents

1. Introduction	1
2. Vectorization of DGR	2
2.1 Overview of DGR	2
2.2 Correspondence to FORTRAN77EX Compiler	2
2.3 Vectorization	4
2.4 Effects of Vectorization	7
2.5 Environments for Execution	8
3. Vectorization of STREAM V3.1	49
3.1 Basic Equations	49
3.2 Vectorization	52
3.3 Results of Vectorization	54
4. Vectorization of Cella	58
4.1 Dynamic Analysis	58
4.2 Vectorization Method	59
4.3 Vectorization of Computing Result	61
4.4 Effect of Vectorization	61
4.5 Summary	62
5. Improvement of GGR	77
5.1 Overview of GGR	77
5.2 Correspondence to FORTRAN77EX Compiler	77
5.3 Vectorization	78
5.4 Effects of Vectorization	81
5.5 Environment for Execution	82
5.6 Summary	82
6. Concluding Remarks	118
Acknowledgements	118
References	119

1. はじめに

日本原子力研究所情報システムセンターは、スーパーコンピューターの有効利用及び利用技術の向上のため、原子力コードのベクトル化及び高速化のための改良を行ってきた [1], [2]。平成5年度にベクトル化及び高速化を行った原子力コードは、分子動力学コード DGR, 3次元非定常圧縮性流体解析コード STREAM V.3.1, 2次元セル・オートマトン流体解析コード Cella 及び分子動力学コード GGR である。

分子動力学コード DGR はダイヤモンド型結晶において放射線による損傷をシミュレーションするコードであり、原子間力の計算においては中心力だけでなく非中心力も考慮され、より現実的なシミュレーションが可能になっている。オリジナル DGR コードは、ベクトル計算機に対するチューニングは施されておらず、粒子数の増大やより長時間の計算の必要性からベクトル化による高速化が期待されていた。今回の作業では主に粒子に関する DO ループのベクトル化を進め、2.8 倍（対 VP2600 スカラー計算）の速度向上率を得た。

3次元非定常圧縮性流体解析コード STREAM V.3.1 は非圧縮性流体を取り扱っていた前バージョンを改良して、圧縮性流体も取り扱えるようにした流体シミュレーションコードである。今回の作業においては主にソルバー部分にハイバープレーン法を適用してベクトル化を進めた。その結果 6.8～14.8 倍（対 VP2600 スカラー計算）の速度向上率を得た。

2次元セル・オートマトン流体解析コード Cella は六角格子上のブランチと格子点に各々 1 ビットの仮想粒子を置き、一時間ステップ毎に粒子を一格子距離だけ進め、その際粒子間の衝突は反応テーブルに従って散乱させる 2 次元流体のセル・オートマトンモデルを用いたコードである。今回の作業では主に流れ場の格子点に関するループでのベクトル化を進めた。その結果 15～16 倍（対 VP2600 スカラー計算）の速度向上率を得た。

一方、分子動力学コード GGR は黒鉛型結晶の特徴である共有結合性を考慮するため、上記 DGR コードと同様に原子間力に中心力だけでなく非中心力も含む放射線による照射損傷をシミュレーションするコードである。本オリジナルコードは上記 DGR と同様、ベクトル計算機向きにコーディングされておらず、ベクトル化による効果が期待された。しかし、アルゴリズム自身がベクトル計算に不向きなため一部ベクトル化とコードの倍精度化を行い 1.23 倍（対 VP2600 スカラー計算）の速度向上率を得た。

以下の章では、これらのコードの概要、ベクトル化の手法、ベクトル化による効果、計算結果の評価及びコードの改良手法を詳しく示す。

2. DGR のベクトル化

ダイヤモンド型結晶における照射損傷をシミュレーションする分子動力学コード DGR [3] のベクトル化を行った。DGR コードでは、ダイヤモンド型結晶に特徴的な共有結合を考慮し、原子間相互作用として、中心力の他に非中心力を取り入れることによって、より現実的な結晶構造におけるシミュレーションが可能である。しかし、DGR コードはベクトル計算機向きにコーディングされていないため、ベクトル計算機による実行時間の短縮は得られない。今回のベクトル化により、ベクトル化版の VP2600/10 (以下 VP2600) 上のベクトル実行時間は、オリジナル版の M780/20 (以下 M780) の実行に比べて、14.2 倍、VP2600 上のスカラ実行に比べて 8.2 倍という高速化がなされた。これにより、多数の粒子、および長時間のシミュレーションが可能になった。

2.1 コード概要

DGR コードはダイヤモンド型結晶構造を持つ物質の照射損傷をシミュレーションするための分子動力学コードであり、PART1、PART2、および PART3 からなっている。PART1 では、結晶とその初期条件の設定、原子間力やポテンシャル関数のテーブル化、および相互作用する隣接原子を検索するテーブルの設定等を行う。PART1 の流れを Fig. 2.1 に示す。

PART2 では PART1 で設定されたデータファイルを読み込んで、結晶を構成する各粒子に対する運動方程式を解き、種々のエネルギーについて計算する。中心力の他に、共有結合における変角力を模擬するための非中心力作用を扱っている点が特徴である。PART2 の流れを Fig. 2.2 に示す。

PART3 では、PART2 で出力された計算結果を図形出力する。

2.2 FORTRAN77EX への対応

前節のとおり、DGR コードは PART1 から PART3 までの 3 段階から成り立っている。これらのコードは FORTRAN-HE コンバイラの文法でコーディングされており、現在使用されている FORTRAN77EX コンバイラではコンパイルできない。このため FORTRAN77EX コンバイラに対応させるために変更を行った。

2.2.1 代入文の変更

FORTRAN-HE では以下の例に示すように一つの代入文で複数の変数を定義することができます。しかし、FORTRAN77EX ではこれができないため、分割した。修正したサブルーチンは GR2D10(PART1), GR2D08(PART2) である。

例

```
IUNT15 = IUNTS = 5 → IUNT15 = 5
                           IUNTS = 5
```

2.2.2 組込み関数の変更

DGR コードでは、いくつかの組込み関数を参照しているが、その中に FORTRAN77EX の組み込み関数と名前が異なるものがある。このため組み込み関数名を変更した。修正したサブルーチンは GR2D41(PART2) である。

変更した組み込み関数

ARCOS → ACOS

2.2.3 引数としての文字の扱いの変更

副プログラムの呼び出しで、引数に文字を直接指定する場合、FORTRAN-HE と FORTRAN 77EX とでは扱い方が異なる。次に示すように変更した。修正したサブルーチンは XYPOUT(PART3) である。

文字の扱いの変更

```
CALL SYMBOL(0.0DO, -25.0DO, 4.0DO, 0ZMIN=0, 0.0DO, 5)
      ↓
CALL SYMBOL(0.0DO, -25.0DO, 4.0DO, 'ZMIN=', 0.0DO, 5)
```

2.2.4 プログラム単位に関する宣言文の変更

メインプログラムに名前を付けるとき、FORTRAN-HE では ELEMENT 文を使用する。しかし、JIS 規格の FORTRAN 文法では、PROGRAM 文を用いると規定されている。このため ELEMENT 文から PROGRAM 文に変更する。変更を行ったサブルーチンは GR1D00(PART1), GR2D08(PART2), GR3D00(PART3) である。

宣言の変更

ELEMENT GR2D08 → PROGRAM GR2D08

2.2.5 倍精度化

DGR コードの FORTRAN-HE 版では、コンパイル時にコンパイルオプション 'AUTODBL' を指定して倍精度化を行っていた。FORTRAN77EX にも AUTODBL や DOUBLE のオプションはあるが、今回の改良でコード自体を倍精度化した。

倍精度化の変更では変換ツール STREAM77 を使用して、各サブルーチンに IMPLICIT REAL*8 (A-H,O-Z) を挿入することと、実定数の倍精度化を行った。この変更は PART1, PART2, PART3 の全のサブルーチンに対して行われた。

倍精度化の変更例

SUBROUTINE ABC(XX,YY)	SUBROUTINE ABC(XX,YY)
YY = XX * 3.14	→ IMPLICIT REAL*8 (A-H,O-Z)

```

RETURN                         YY = XX * 3.14D0
END                           RETURN
                               END

```

2.3 ベクトル化

DGR コードを構成する PART1, PART2, および PART3 で, 今回の入力データを用いた実際の CPU 時間はそれぞれ 0.2 秒, 14.2 秒, 3.1 秒 (全て VP2600 上) である. ベクトル化を行ったコードは, この中で計算時間が最も長い PART2 である. PART2 は結晶を構成する各粒子に対する運動方程式を解き, 種々のエネルギーを計算する. このため, 粒子の数を多くしたり, 解析時間を長くすると, 計算時間はさらに長くなる.

PART2 の処理を見ると, 各粒子については独立に計算できる構造になっているため, ベクトル化の方針としては粒子数に関するベクトルループが出来るように処理を変更する. この結果, 粒子数が多くなるほどベクトル長が長くなるため, より高い速度向上率が期待される.

2.3.1 動的挙動解析

PART2 の実際のコスト分布を調査するため, 動的挙動解析ツール ANALYZER [11] を用いた解析を行った. その結果を Table 2.1 に示す. この表によるとサブルーチン GR2D41, GR2D4A の 2 つで全コストの 93 % を越え, また実行回数も非常に多いので, これらをベクトル化の対象とする.

Fig. 2.2 に示した PART2 の計算の流れの中で, サブルーチン GR2D3 から GR2D60 に至る大粒度のループが存在するが, これは時間ステップに関するループのためベクトル処理できない. GR2D40 の中には粒子のループがあり, このループ内で各粒子に対する運動方程式を解く. このループをベクトル処理する. 今回使用した入力データでは粒子数 620 であり, 十分に長いベクトル長が得られるため, 高速化が期待できる. 将来的には, 約 20000 個の粒子を用いたシミュレーションを行いたいので, さらに高い速度向上率が得られることになる. またこのループ内では GR2D41, さらに GR2D41 からは GR2D4A が呼び出されているので, これら 2 つの呼び出し回数が多く, オーバーヘッドがかかる. これらの調査結果を基にベクトル化作業を進めた.

2.3.2 ベクトル化のための変更

GR2D40 は各粒子についての計算を行う部分であり, PART2 全体の処理の中で中心的な部分である. オリジナル版では, PART1 の出力結果より, 各粒子の初期条件 (座標等) を入力し, それを元に自分自身の粒子と回りに近接した結合相手である粒子の影響を考慮して, 物性値の計算を行う. すべての粒子について計算が終わった後, 粒子の新しい位置 (座標) を計算し, 再び物性値の計算を進めていく.

近接した粒子との相互作用は, 「箱検索」という方法を用いて計算される. これは粒子をいくつかの箱に入れて, インデックスで指示した箱に対して, その箱に属する粒子との相互作用を計算するものである. DGR では 1 つの箱の中に 6 つまでの粒子が入る. 計算の対象となる粒子は, 粒子自身を含む箱とその回りに近接する箱 26 個 (計 27 個, ただし境界部分は境界条件が設

定されるので、少なくなる)に含まれる粒子との相互作用を計算するようになっている (Fig. 2.3 参照)。また、箱に存在する粒子は時間がたつにつれて他の箱に移動する場合がある。

コード上ではサブルーチン GR2D40 で粒子に関するループがあり、その中で、GR2D41 を呼び出している。GR2D41 では加速度や位置エネルギー等を求める。GR2D41 の中では、回りの箱を検索するループがあり、その中に箱内の粒子を検索するループが存在する。そこで、ベクトル化対象となるループは粒子のループで行い、箱内、および回りの箱のループと入れ換えることが必要である。

2.3.2.1 サブルーチン GR2D40

GR2D40 は粒子ループ (D0 5000) 内の先頭で GR2D41 の呼び出しがあり、引数として粒子番号を渡している。以下の処理では、GR2D41 で計算された種々のパラメータを使って、位置エネルギーや加速度の計算を行っている。GR2D41 の中でベクトル化するために粒子ループを引き込み、GR2D41 の呼び出し時の引数に粒子数 N を指定した。加速度 AC は全ての粒子について一度に GR2D41 で計算するが、GR2D41 の呼び出し以降の処理で AC を使用するので、全ての粒子の加速度を保存しておくために、配列 AC の次元を上げた ($AC(3) \rightarrow AC(4000, 3)$)。

呼び出し以降の処理では、粒子ループについて回帰計算などのベクトル処理不可能な部分がないので、ベクトル処理が可能だが、コーディングが複雑なためベクトル処理が一部できない。この部分の処理は配列 IBND (境界パラメータ) の値によって 3 種類の処理を行っている。そこで、その種類ごとのリストを作り、それぞれ単純なループで処理することにした。ただし 3 種類の内 1 つは、リスト生成時に同時に計算することができる所以リストによる処理は 2 種類になる。すなわち IBND が正の値の場合と、負の場合の 2 種類を作り、それぞれのループで計算を行う。Fig. 2.4 に GR2D40 のオリジナルプログラム、Fig. 2.5 にベクトル化版のプログラムを示す。

2.3.2.2 サブルーチン GR2D41, GR2D4A, GR2D3

GR2D41 の処理の概要を Fig. 2.6 に示す。オリジナル版では、親ルーチンである GR2D40 の粒子ループの中で呼ばれているが、ベクトル化のために粒子ループをこのルーチンに引き込む。

ベクトル化するために GR2D41 を大きく 4 つのブロックに分けた。初期値設定部分の 2 カ所、加速度、および角度計算部分の 2 カ所である。

第 1 ブロックの初期値設定部分では、GR2D41 内で使用される角度、距離等が入る配列 (NF, R 他) のゼロクリアをしており、GR2D41 が呼び出されるたびに実行されなくてはいけない。この部分を粒子ループでベクトル化するが、初期値を設定するためにループ長の短い D0 ループを使っているものがある。多重ループの場合のベクトル化対象ループは原則的に最内ループであることから、粒子ループ全体をベクトル化するために、これらのループ長の短いループは全て展開した。

第 2 ブロックの初期値設定部分は、箱のある位置における粒子数の設定 (配列 NVAL) である。すなわち箱の位置によって (境界条件があるため) 箱に含まれる粒子数が異なる。その定義をするところであるが、この部分は最初の 1 回のみ計算すれば済むところなので、フラグを設定して GR2D41 が最初に実行されるときのみ計算するようにした。

第 3 ブロックの加速度、および角度計算部分はオリジナル版では ICOMB=0 から文番号 7500

のCONTINUE文までである。この処理には箱内のループ長6と27個の箱の2重ループがある。いずれもIF文とGO TO文によるループであるが、これをDOループに置き換えてベクトル化処理を行っても、ベクトル長が短いために高速化は望めない。箱内、箱ごとのループと上位ルーチンから引き込んだ粒子ループを反転させれば、粒子ループのループ長は今回の入力データでは630で高速化が望める。しかし、単純に反転させることはできない。これは入力データで指定され結合手に強制的に連結される粒子が存在するからである。この粒子に対しては、箱内、および回りの箱に関する2つのループによらない繰り返し計算を行う。このため、リスト方式によるベクトル処理を行うことにした。

リスト作成部分は大きく2つに分かれる。前半は強制的に結合された粒子に対するリストの作成部分であり、後半は27個の箱、および箱内のループに対するリスト作成である。オリジナルでの粒子の処理順番を粒子番号で示すと

1,1,1, . . . , 2,2,2, . . . , 3,3,3, . . . , 619,619,619, . . . , 620,620,620

となる。すなわち、同一の粒子に対し、箱内のループ×回り箱のループ(6×27)分の繰り返しによる処理が行われる。このままリストに収めて計算を行うと、位置エネルギーの計算部分で総和計算があるため回帰計算が発生する。このため

1,2,3, , 619,620	(1ベクトル処理)
1,2,3, , 619,620	(1ベクトル処理)
•	
•	
1,2,3, , 619,620	(1ベクトル処理)

のように分割して2次元配列のリストに収納する。ただし、このまま配列宣言をすると、粒子同士が強制的に連結される分のリストを含めて、リストの配列の大きさは、(粒子数, $27*6+1$)となる。リストにすべき変数はいくつかあり、粒子数が多くなると極めて使用されるメモリが大きくなってしまう。しかし、実際の計算では、粒子が無いときには計算部分をスキップするようになっている。そこで粒子があるときだけリストに格納することにする。今回の入力データでは、 $27*6+1$ は必要なく30で十分である。ただし、入力データによっては大きくなることもあるので、大きさのチェックをリスト作成時に行い、もし宣言した大きさを越える場合にはメッセージを出力して実行をストップするようにしてある。この場合は、配列の大きさをPARAMETER文によって変数LV用いて定義しており、LVの値を変えれば、全てのリストの大きさが変わるようにになっている。リストにしたものは粒子番号(I1、リスト名 I1L)、影響の対象となる粒子(IBOX、リスト名 IBTBL)、強制的に結合された粒子がある場合のフラグ(ICOMB、リスト名 ICOM¥)、強制的に連結された粒子間との距離のチェック用フラグ(NAL、リスト名 NAL¥)である。また、各ベクトル処理毎のループ長を配列ICNTに格納した。リスト作成後、加速度、および角度計算の前半部分をリストベクトル方式によってベクトル処理を行う。角度Rの計算部分では上向きGO TO文による繰り返しループがあるが、書き下すことで上向きGO TO文を無くした。

リストは GR2D41 を実行する毎に作成する必要はない。各粒子の位置が変わった時に行えばよい。位置が変わることとは、配列 IBOX の中の粒子が移動することである。IBOX の定義部分はサブルーチン GR2D3 で行われる。この一部を Fig. 2.7 に示す。座標は時間ループに関して毎回計算され、その座標の値によって箱である IBOX に、ある計算式に従って格納される。座標は毎回変化していくが、この計算式によると、

```
IX = X(I1,1)*0.55D0
IY = Y(I1,1)*0.55D0
IZ = Z(I1,1)*0.55D0
I2 = 1+IX+IALF*IBE*IZ
```

ただし、X, Y, Z は実数型、その他はすべて整数型

であり、I2 が IBOX の添字となる。この式では座標値の整数部分のみ取り出して計算することになるので、毎回 I2 の値が変わることは非常に少ない。そこで、I2 の計算時に一回前の値と比較し、異なる場合があったときのみリストを再作成することにした。同様に、GR2D3 での IBOX への代入もこのときに行う。この部分を Fig. 2.8 に示す。

第4ブロックの加速度、および角度計算は粒子ループに関して行う。ただし、内部には小さな D0 ループがいくつか存在するため、全体のベクトル化に弊害を及ぼす。このため、すべて展開した。また、ここからサブルーチン GR2D4A を呼び出している。CALL 文はベクトル化非対象であるが、FORTRAN77EX では INLINE 展開することでベクトル化が可能である。そこで、コンパイル時に OPT(F) を指定して、INLINE 展開させることにした。これで後半部分のベクトル処理が可能になった。オリジナルの GR2D41 を Fig. 2.9 に、ベクトル化版の GR2D41 を Fig. 2.10 に示す。

2.4 ベクトル化の効果

ここでは、ベクトル化版の計算結果、および効果について述べる。

2.4.1 計算結果

作成したベクトル化版を用いて計算を行った。得られた計算結果はオリジナルとすべての桁で一致した。

2.4.2 ベクトル化の効果

ベクトル化版に対し ANALYZER を用いてコスト分布を調査した。その結果を Table 2.2 に示す。GR2D41 では下位ルーチンの GR2D4A を引き込んだので、コスト比率が上昇している。しかし、GR2D41 は大部分がベクトル処理可能である。

また、2つの入力データを用いてベクトル化版の速度向上を測定した。この結果を Table 2.3 に示す。この表によれば、入力データ TYPE1 で VP2600 上でオリジナルスカラに対してベクトル化は約 8.2 倍、M780 上のオリジナル版との比較では 14.2 倍の倍率が達成された。また、TYPE 2 では同様に 8.5 倍、14.7 倍になった。粒子数を多くしたり、解析時間を長くすれば、さらに速度向上率が高くなるはずである。

2.5 実行環境

ここではベクトル化版の DGR コードを実行する際の実行環境について述べる。

2.5.1 データセット関係

DGR コードのオリジナル版は FORTRAN-HE による単精度のコードであった。今回の作業によって、FORTRAN77EX 仕様の倍精度コードを作成し、さらにベクトル化版を作成した。これらのデータセットを Table 2.4 に示す。

2.5.2 ロードモジュール作成、および実行方法

2.5.2.1 ロードモジュール作成

Fig. 2.11 に DGR ベクトル化版ロードモジュールを作成する JCL を示す。ベクトル化版のロードモジュールを作成する場合サブルーチン GR2D4A を内部展開させる必要があるため、コンパイルオプションに OPT(F) を指定する必要がある。

2.5.2.2 実行

実行方法は VP で実行するための CLASS 指定することを除いて、今までの実行とほとんど変更はない。実行 JCL を Fig. 2.12 に示す。

2.5.2.3 実行時の注意

ベクトル化版ではベクトル化のために必要となる作業領域をコード内で確保している。入力データによってはこの作業領域が不足する場合がある。もし不足した場合は、以下のメッセージが出力されジョブは終了するようになっている。

PLEASE INCREASE LV-VALUE IN GR2D41

この場合はサブルーチン GR2D41 の PARAMETER 文で宣言されている LV の値を大きくすればよい。この宣言部分を Fig. 2.13 に示す。

また解析の対象となるデータで、粒子数を多くする場合もベクトル化作業領域を大きくする必要がある。オリジナルでは、解析可能な最大粒子数は 4000 であり、コード内で使用されている配列もすべて大きさ 4000 で宣言されている。粒子数を多くする場合は、これらの配列を変更するとともに、ベクトル化作業用の配列も変更する必要がある。Table 2.5 にベクトル化版に追加された配列を示す。

Table 2.3 CPU Time table of the Original and Vectorized Versions

項目		入力データ	TYPE1	TYPE2
CPU 時間	オリジナル(M780)	24s72	49s38	
	オリジナル(VP2600)	14s28	28s41	
	VP版スカラモード	14s34	28s11	
	VP版VPモード (VU時間)	1s74 (1s30)	3s36 (2s55)	
使用領域 (KB)	オリジナル(M780)	1104	1104	
	オリジナル(VP2600)	1296	1296	
	VP版スカラモード	4132	4132	
	VP版VPモード	4432	4432	
I/O 回数	オリジナル(M780)	575	1058	
	オリジナル(VP2600)	574	1057	
	VP版スカラモード	574	1066	
	VP版VPモード	574	1066	
倍率	オリジナル(M780)/VP版VPモード	14.20	14.70	
	オリジナル(VP2600)/VP版VPモード	8.21	8.46	
	VP版スカラモード/VP版VPモード	8.24	8.37	
ベクトル化率(%)		96.9	97.1	

Table 2.4 Datasets of DGR Source and JCL

バージョン	データセット名	名前	補足
オリジナル版	PART1	J1476.DDGR1.FORT77	FORTRAN-HE仕様 単精度版
	PART2	J1476.DDGR2.FORT77	
	PART3	J1476.DDGR3.FORT77	
オリジナル版	PART1	J0001.DDGR1D.FORT77	FORTRAN77EX 仕様 倍精度版
	PART2	J0001.DDGR2D.FORT77	
	PART3	J0001.DDGR3D.FORT77	
ベクトル化版	PART2	J0001.DDGR2DVP.FORT77	FORTRAN77EX 倍精度
J C L	PART1	J0001.DGRVP.CNTL	PART1 翻訳から実行まで
	PART2		FORTCL ロードモジュール作成
			GO 実行
	PART3		PART3 翻訳から実行まで

Table 2.5 Added or Modified Arrays

サブルーチン名	ベクトル化版での宣言	旧版での宣言
G R 2 D 3	I2¥(4000)	なし
G R 2 D 4 A	R1(4000,3) R2(4000,3) R3(4000,3) AC(4000,3)	R1(3) R2(3) R3(3) AC(3)
G R 2 D 4 0	AC(4000,3)	AC(3)
G R 2 D 4 1	AC(4000,3) J1¥(4000) I2¥(4000) IC¥(4000) NVAL¥(4000) IBTBL(4000,LV)* I1L(4000,LV)* NAL¥(4000,LV) * ICOMP¥(4000,LV)* ICNT (LV) * NF(4000,4) RNF(4000,4) THETA(4000,6) R(4000,3,4)	AC(3) なし なし なし なし なし なし なし なし なし なし NF(4) RNF(4) THETA(6) R(3,4)

*注意 LVは PARAMETER文で定義している。(LV=30)

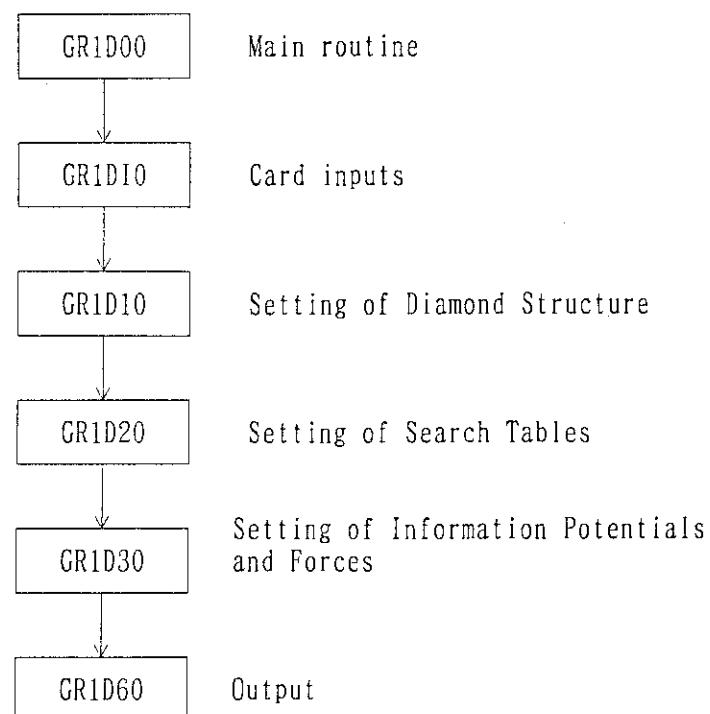


Fig.2.1 Flow Chart of the PART1 of the DGR Code

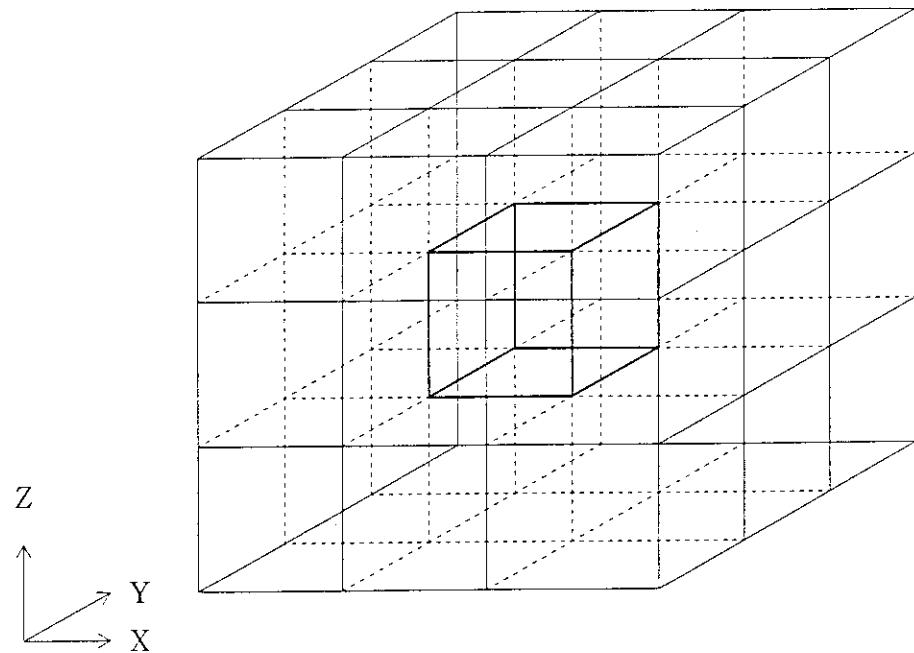


Fig.2.3 Method of Searching Boxes

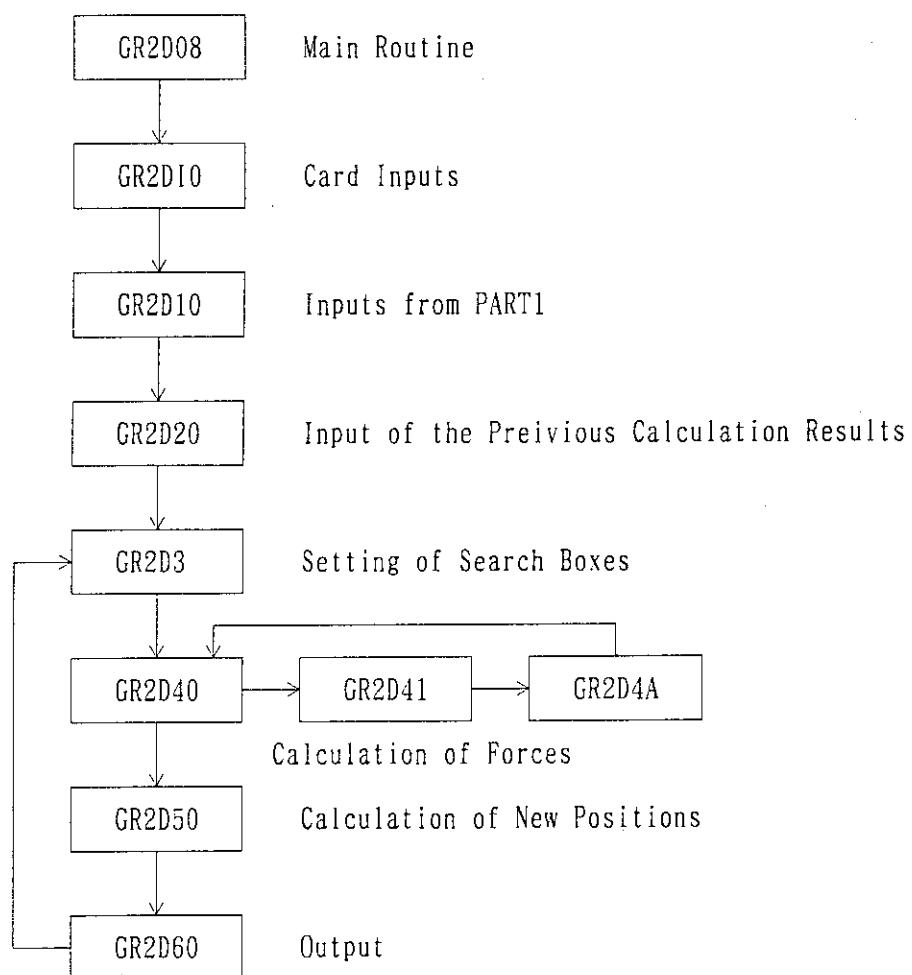


Fig.2.2 Flow Chart of the PART2 of the DGR Code

```

#####
## GR2D40 ####
## SSN=0001
## SSN=0002
## SSN=0003

      FILE=J0001.DDGR2D.FORT77(GR2D40)          FORTRAN 77/VP V12L10
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      SUBROUTINE GR2D40
      IMPLICIT REAL*8 < A-H,O-Z >
      C      COMMON /COMBIN/KK,LL11(50),LL14(50)
      C
      1. FUNCTION
      C      (1) FOR DIAMOND TYPE OF PART-2
      C      (2) CALCULATE DISSIPATION ENERGY,SPRING ENERGY,PRESSURE ENERGY,
      C      ACCELERATION,PARAMETER OF MOTION
      C
      2. INPUT
      C      NUMBER OF PARTICLES           SEC
      C      DDF                           DT**2
      C      TB                            INITIAL COORDINATES
      C      (4000,3)
      C      IBND                         BOUNDARY TYPE
      C      (4000)
      C      X                             COORDINATES
      C      (4000,3)
      C      VX                            PARAMETER OF MOTION
      C      (4000,3)
      C      AC(3)                         ACCELERATION
      C      AC(26,3)                      PRESSURE CONSTANT OF THE FIRST LAYER
      C      BC(26,3)                      SPRING CONSTANT
      C      CC(26,3)                      VISCOS FORCE CONSTANTS
      C      C(26,3)                       OF THE SECOND LAYER
      C      A2(26,3)
      C      B2(26,3)
      C      C2(26,3)
      C      C2(26,3)
      C      ENGL                          DISSIPATION ENERGY
      C      ENGSP                         SPRING,PRESSURE ENERGY
      C
      3. OUTPUT
      C      VX                            PARAMETER OF MOTION
      C      (4000,3)
      C      ENGL                          DISSIPATION ENERGY
      C      ENGSP                         SPRING,PRESSURE ENERGY
      C
      4. LOCAL VARIABLES
      C      I                            PARTICLE NUMBER
      C
      5. CALLED BY GR2D08
      C      CALLS                         GR2D41
      C      7. RESTRICTION
      C      8. REFERENCE
      C
      DIMENSION TB(4000,3),IBND(4000),
      -          X(4000,3),VX(4000,3),
      -          AC(3),
      -          AC(26,3),B(26,3),C(26,3),
      -          A2(26,3),B2(26,3),C2(26,3),
      -          IC0V(4000)
      C
      COMMON /CNTRL/CNTRL(29),
      -          /GEOM2/GEOM2(4,3,377),
      -          /DYNA/DYNA(4,24,98),
      -          /WORK/WORK(800,6),
      C
      SSN=0004
      SSN=0005

```

Fig.2.4 Subroutine GR2D40

```

★ GR2D40 < 2 >

      FILE=J0001.DDGR2D.FORT77(GR2D.O)          FORTRAN 77/VP V12L10
      C-----1-----2-----3-----4-----5-----6-----7-----8
      EQUIVALENCE (CNTRL(6),N),
      EQUIVALENCE (CNTRL(9),DDT),
      EQUIVALENCE (GEOM1(13),TB(1,1)),
      EQUIVALENCE (GEOM2(12013),IBND(1)),
      EQUIVALENCE (DYNA(1),X(1,1)),
      EQUIVALENCE (DYNA(12001),VX(1,1)),
      EQUIVALENCE (DYNA(24001),ACC(1)),
      EQUIVALENCE (DYNA(42016),A(1,1)),
      EQUIVALENCE (DYNA(42094),B(1,1)),
      EQUIVALENCE (DYNA(42172),C(1,1)),
      EQUIVALENCE (DYNA(42250),A2(1,1)),
      EQUIVALENCE (DYNA(42328),B2(1,1)),
      EQUIVALENCE (DYNA(42406),C2(1,1)),
      EQUIVALENCE (DYNA(42485),ENGL),
      EQUIVALENCE (DYNA(42489),ENGSP),
      EQUIVALENCE (WORK(4007),ICOV(1))          PREPARATION
      C-----900 CONTINUE                           * MAIN LOOP
      C-----DO 900 I=1,N                          GET ACCELERATION AND POTENTIAL
      C-----ICOV(I)=0                            ENERGY
      C-----10=IBND(I)
      C-----IF(I<10.GE.1.AND.I0.LE.6) ICOV(I)=2
      C-----IF(I<10.GE.7.AND.I0.LE.10) ICOV(I)=3
      C-----IF(I<10.GE.17.AND.I0.LE.19) ICOV(I)=3
      C-----IF(I<10.GE.19) ICOV(I)=3
      C-----IF(I<10.GE.20.AND.I0.LE.22) ICOV(I)=4
      C-----IF(I<10.GE.23.AND.I0.LE.25) ICOV(I)=3
      C-----IF(I<10.GE.26) ICOV(I)=4
      C-----900 CONTINUE
      C-----DO 5000 I=1,N
      C-----C
      C-----CALL GR2D41(I)
      C-----IF(IBND(I)>1000,3000,2000          FOR SECOND LAYER
      C-----1000 L=-IBND(I)
      C-----DO 1500 J=1,3                          DISSIPATION ENERGY
      C-----ENGL=ENGL-C2(L,J)*(VX(1,J)-X(1,J))*2
      C-----DXYZ=X(1,J)-TB(1,J)                  SPRING, PRESSURE ENERGY
      C-----ENGSP=ENGSP-A2(L,J)*DXYZ-0.5D0*B2(L,J)*DXYZ**2
      C-----ACC(J)=ACC(J)+A2(L,J)*B2(L,J)*(VX(1,J)-X(1,J))
      C-----3-1500 GO TO 3000                   ACCELERATION
      C-----2000 L=IBND(I)                         FOR FIRST LAYER
      C-----3-DO 2500 J=1,3                        DISSIPATION ENERGY
      C-----ENGL=ENGL-C(L,J)*(VX(1,J)-X(1,J))*2
      C-----DXYZ=X(1,J)-TB(1,J)                  SPRING, PRESSURE ENERGY
      C-----3
      SSN=0006
      SSN=0007
      SSN=0008
      SSN=0009
      SSN=0010
      SSN=0011
      SSN=0012
      SSN=0013
      SSN=0014
      SSN=0015
      SSN=0016
      SSN=0017
      SSN=0018
      SSN=0019
      SSN=0020
      SSN=0021
      SSN=0022
      SSN=0023
      SSN=0024
      SSN=0025
      SSN=0026
      SSN=0027
      SSN=0028
      SSN=0029
      SSN=0030
      SSN=0031
      SSN=0032
      SSN=0033

```

Fig.2.4 Subroutine GR2D40 (continued)

```

★ GR2D40 < 3 >

      FILE=J00001.DDGR2D.FORT77(GR2D40)
      C----*---1----*---2----*---3----*---4----*---5----*---6----*---7----*---8
      3   ENGSP=ENGSP-A(L,J)*DXYZ*0.5DO*B(L,J)*DXYZ*2
      C   ACCELERATION
      1+-----3-2500 ACC(J)=ACC(J)+ACC(L,J)+B(L,J)*DXYZ+C(L,J)*VX(L,J)-X(I,J)
      C
      1     3000 CONTINUE
      C   PARAMETER OF MOTION
      1     2-----3-----DO 3500 J=1,3
      1+-----3-----3500 VX(I,J)=2.D0*(ACC(J)*DDT+VX(I,J))-X(I,J)
      C
      1     +-----5000 CONTINUE
      C
      SSN=0039
      SSN=0040
      SSN=0041

```

Fig.2.4 Subroutine GR2D40 (continued)

```

***** GR2D40 ***** FILE=J0001.DDGR2DVP.FORT77(GR2D40) FORTRAN 77/VFP V12L10
C-----1-----2-----3-----4-----5-----6-----7-----*-----8-----*
      SUBROUTINE GR2D40
      IMPLICIT REAL*8 ( A-H,O-Z )
      GR2D40      5.0
      COMMON /COMBIN/VKK,I111(50),I114(50)
      SSN=0003
      C 1. FUNCTION
      C (1) FOR DIAMOND TYPE OF PART-2
      C (2) CALCULATE DISSIPATION ENERGY,SPRING ENERGY,PRESSURE ENERGY,
      C      ACCELERATION,PARAMETER OF MOTION
      C 2. INPUT
      C      SEC
      C      N      NUMBER OF PARTICLES
      C      DOT     DT**2
      C      TB      INITIAL COORDINATES
      C      (4000,3)
      C      IBND   BOUNDARY TYPE
      C      (4000)
      C      X      COORDINATES
      C      (4000,3)
      C      VX      PARAMETER OF MOTION
      C      (4000,3)
      C      AC(3)  ACCELERATION
      C      AC(26,3) PRESSURE CONSTANT OF THE FIRST LAYER
      C      AC(26,3) SPRING CONSTANT
      C      BC26(3) SPRING CONSTANT
      C      C(26,3) VISCOS FORCE CONSTANTS
      C      A2(26,3) OF THE SECOND LAYER
      C      B2(26,3)
      C      C2(26,3)
      C      C2(26,3) DISSIPATION ENERGY
      C      ENGL   SPRING,PRESSURE ENERGY
      C      ENGSP
      C 3. OUTPUT
      C      VX      PARAMETER OF MOTION
      C      (4000,3)
      C      ENGL   DISSIPATION ENERGY
      C      ENGSP SPRING,PRESSURE ENERGY
      C 4. LOCAL VARIABLES
      C      I      PARTICLE NUMBER
      C 5. CALLED BY GR2D08
      C 6. CALLS GR2D41
      C 7. RESTRICTION
      C 8. REFERENCE
      C
      C      DIMENSION TB(4000,3),IBND(4000),
      C                  X(4000,3),VX(4000,3),
      C                  ACC(4000,3),
      C                  A(26,3),B(26,3),C(26,3),
      C                  A2(26,3),B2(26,3),C2(26,3),
      C                  IC0V(4000),
      C                  AC(3),
      C      COMMON /VEC1/AC
      C
      SSN=0004
      C
      SSN=0005
      C
      SSN=0006
      C      COMMON /CNTRL/CNTRL(29),
      C                  /GEOM2/GEOM2(43377),

```

Fig.2.5 Modified Subroutine GR2D40

```

★ GR2D4O ( 2 )

C----*-1-----*--2-----*---3-----*---4-----*---5-----*---6-----*---7-----*---8-----*
      FILE=J0001-DDGR2DV.P.FORT77(GR2D4O)  FORTRAN 77/VP V12L10
      /DYNA(DYNA(42498),
      /WORK(WORK(8006))

      SSN=0007
      SSN=0008
      SSN=0009

      C     EQUIVALENCE (CNTRL(6),N),
              (CNTRL(9),DDT),
              (GEOM2(13),TB(1,1)),
              (GEOM2(12013),IBND(1))
      EQUIVALENCE (DYNAC(1),X(1,1)),
              (DYNAC(12001),VX(1,1)),
              (DYNAC(42016),A(1,1)),
              (DYNAC(4204),B(1,1)),
              (DYNA(42112),C(1,1)),
              (DYNA(42250),A2(1,1)),
              (DYNA(42328),B2(1,1)),
              (DYNA(42406),C2(1,1)),
              (DYNA(42485),ENGL),
              (DYNA(42489),ENGSP)
              (DYNA(24001),ACC1),
              (DYNA(24001),ACCV1)

      C.TN  EQUIVALENCE (WORK(4007),ICOV(1))
            DIMENSION LIP(4000),LIM(4000)
                  PREPARATION
      SSN=0010
      SSN=0011
      C     V-----DO 900 I=1,N
              ICOV(I)=0
              IO=IBND(I)
              IF(IO-GE-1-AND-IO.LE.-6) ICOV(I)=2
              IF(IO-GE-7-AND-IO.LE.-18) ICOV(I)=3
              IF(IO-EQ.-19) ICOV(I)=3
              IF(IO-GE.-20-AND-10-LE.-22) ICOV(I)=4
              IF(IO-GE.-23-AND-10-LE.-25) ICOV(I)=3
              IF(IO-EQ.-26) ICOV(I)=4
      900 CONTINUE
      C     DO 5000 I=1,N
                  * MAIN LOOP
      C     CALL GR2D41(N)
      C     GET ACCELERATION AND POTENTIAL
              ENERGY
      SSN=0022
      C     IF(IBND(I)) 1000,3000,2000
                  FOR SECOND LAYER
      C1000 L=-IBND(I)
      C     DO 1500 J=1,3
      C     ENGL=ENGL-C2(L,J)*(VX(I,J)-X(I,J))*2
              DISSIPATION ENERGY
      SPRING,PRESSURE ENERGY
      C.TN DXYZ=X(I,J)-TB(I,J)
      C.TN ENGSP=ENGSP-A2(L,J)*DXYZ*2*(L,J)*DXYZ**2
              ACCELERATION
      C1500 ACC(J)=ACC(J)+A2(L,J)*DXYZ+C2(L,J)*(VX(I,J)-X(I,J))
              GO TO 3000
      C.TN L=IBND(I)
      C     DO 2500 J=1,3

```

Fig.2.5 Modified Subroutine GR2D4O (continued)

```

★ GR2D40 ( 3 )

C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C-----*-----FILE=J0001.DDGRR2DVP.FORT77(GR2D40) FORTRAN 77/VP V12L10
C-----*-----DISSIPATION ENERGY
C-----*-----SPRING/PRESSURE ENERGY
C-----*-----TN DXYZ=X(I,J)-TB(I,J)
C-----*-----TN ENGL=ENGL-C(L,J)*(VX(I,J)-X(I,J))**2
C-----*-----AC(J)=AC(J)+A(L,J)+B(L,J)*DXYZ**2
C-----*-----AC(J)=AC(J)*DDT+VX(I,J)*VX(I,J)-X(I,J)
C-----*-----CONTINUE
C-----*-----PARAMETER OF MOTION
C-----*-----TN DO 3500 J=1,3
C-----*-----3500 VX(I,J)=2.0*(AC(J)*DDT+VX(I,J))-X(I,J)
C-----*-----CONTINUE
C-----*-----M = 0
C-----*-----MM = 0
C-----*-----DO 5001 I = 1, N
C-----*-----IF(IBND(I).GT.0) THEN
C-----*-----FOR FIRST LAYER - LIST
C-----*-----M = M + 1
C-----*-----LIP(M) = I
C-----*-----ELSE IF(IBND(I).LT.0) THEN
C-----*-----FOR SECOND LAYER - LIST
C-----*-----MM = MM + 1
C-----*-----LIM(MM) = 1
C-----*-----ELSE IF(CIBND(I).EQ.0) THEN
C-----*-----PARAMETER OF MOTION
C-----*-----TN DO 4500 I = 1, 1
C-----*-----VX(I,1)=2.0*(ACC(I,1)*DDT+VX(I,1))-X(I,1)
C-----*-----VX(I,2)=2.0*(ACC(I,2)*DDT+VX(I,2))-X(I,2)
C-----*-----VX(I,3)=2.0*(ACC(I,3)*DDT+VX(I,3))-X(I,3)
C-----*-----END IF
C-----*-----5001 CONTINUE
C-----*-----FOR SECOND LAYER
C-----*-----VOCCL LOOP NOVREC
C-----*-----DO 5002 LL = 1, MM
C-----*-----I = LIM(LL)
C-----*-----L = -IBND(I)
C-----*-----ENGL=ENGL-C2(L,1)*(VX(I,1)-X(I,1))**2
C-----*-----DXYZ=X(I,1)-TB(I,1)
C-----*-----ENGSP=ENGSP-A2(L,1)*DXYZ-0.5DO*B2(L,1)*DXYZ**2
C-----*-----ACC(I,1)=ACC(I,1)+B2(L,1)*DXYZ+C2(L,1)*(VX(I,1)-X(I,1))
C-----*-----ENGL=ENGL-C2(L,2)*(VX(I,2)-X(I,2))**2
C-----*-----DXYZ=X(I,2)-TB(I,2)
C-----*-----ENGSP=ENGSP-A2(L,2)*DXYZ-0.5DO*B2(L,2)*DXYZ**2
C-----*-----ACC(I,2)=ACC(I,2)+A2(L,2)*DXYZ+B2(L,2)*DXYZ+C2(L,2)*(VX(I,2)-X(I,2))
C-----*-----ENGL=ENGL-C2(L,3)*(VX(I,3)-X(I,3))**2
C-----*-----DXYZ=X(I,3)-TB(I,3)
C-----*-----ENGSP=ENGSP-A2(L,3)*DXYZ-0.5DO*B2(L,3)*DXYZ**2
C-----*-----ACC(I,3)=ACC(I,3)+A2(L,3)+B2(L,3)*DXYZ+C2(L,3)*(VX(I,3)-X(I,3))
C-----*-----PARAMETER OF MOTION
C-----*-----TN VX(I,1)=2.0*(ACC(I,1)*DDT+VX(I,1))-X(I,1)
C-----*-----VX(I,2)=2.0*(ACC(I,2)*DDT+VX(I,2))-X(I,2)
C-----*-----VX(I,3)=2.0*(ACC(I,3)*DDT+VX(I,3))-X(I,3)
C-----*-----5002 CONTINUE
C-----*-----FOR FIRST LAYER

```

Fig.2.5 Modified Subroutine GR2D40 (continued)

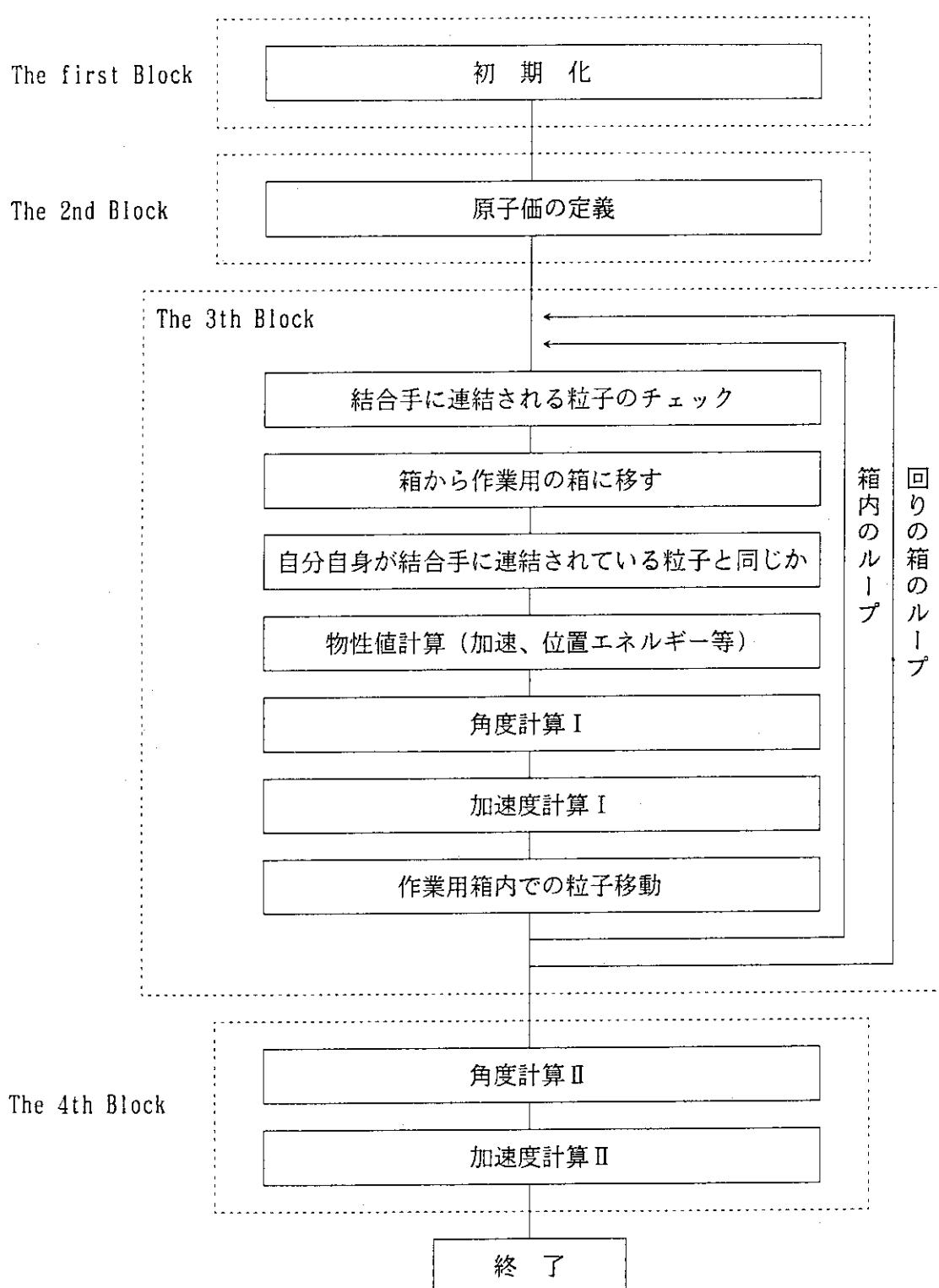


Fig.2.6 Computation Flow of Subroutine GR2D41

```
##### GR2D3 ####
##### SSN=0001
##### SSN=0002
```

```
FILE=J0001.DDGR2D.FORT77(GR2D3)      FORTRAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
SUBROUTINE GR2D3(IER)
IMPLICIT REAL*8 ( A-H,O-Z )
```

```

C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
C-----DO 4000 I1=1,N
C-----CONTINUE
SSN=0009   1          C-----DO 4000 I1=1,N
SSN=0010   1          C     IX=X*(I1,1)+0.55D0
SSN=0011   1          C     IY=X*(I1,2)+0.55D0
SSN=0012   1          C     IZ=X*(I1,3)+0.55D0
SSN=0013   1          C     I2=1+IX+IALP*IY+IBET*IZ
SSN=0014   1          C     IF(I2.GT.IGAM)
SSN=0015   1          C       IF(CIBOX(6,I2).NE.0)
SSN=0016   1          C         IBOX(6,I2)=11
SSN=0017   1          C       IF(CIBOX(5,I2).NE.0)
SSN=0018   1          C         IBOX(5,I2)=11
SSN=0019   1          C       IF(CIBOX(4,I2).NE.0)
SSN=0020   1          C         IBOX(4,I2)=11
SSN=0021   1          C       IF(CIBOX(3,I2).NE.0)
SSN=0022   1          C         IBOX(3,I2)=11
SSN=0023   1          C       CONTINUE
SSN=0024   1          C         IF(CIBOX(2,I2).NE.0)
SSN=0025   1          C         IBOX(2,I2)=11
SSN=0026   1          C       CONTINUE
SSN=0027   1          C         IF(CIBOX(1,I2).NE.0)
SSN=0028   1          C         IBOX(1,I2)=11
SSN=0029   1          C       CONTINUE
SSN=0030   1          C         IF(CIBOX(1,I2).NE.0)
SSN=0031   1          C         IBOX(1,I2)=11
SSN=0032   1          C         CONTINUE
SSN=0033   1          C         IF(CIBOX(1,I2).NE.0)
SSN=0034   1          C         IBOX(1,I2)=11
SSN=0035   1          C         3000  LIBND(I1)=12
SSN=0036   1          C         4000  CONTINUE
SSN=0037   1          C         IER=0
SSN=0038   1          C         GO TO 9000
SSN=0039   1          C         IER=1
SSN=0040   1          C         8100  IER=2
SSN=0041   1          C         9000  CONTINUE
SSN=0042   1          C         9300  CONTINUE
SSN=0043   1          C         RETURN
SSN=0044   1          C         END
SSN=0045   1          C-----CONTINUE
C-----BOX NUMBER
C-----TEST THE CONTENTS OF BOX
C-----BOX NUMBER OVER THE MAXIMUM LIM.
C-----MAIN LOOP ALONG ALL PARTICLES
00004800
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00010700
00010800
00010900
```

Fig.2.7 Subroutine GR2D3

```

# # # # # # # # # #
## GR2D3  ## 
## # # # # # # # # 
SSN=0001
SSN=0002

      C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      FILE=J0001.DDGR2DVP.FORT77(GR2D3)      FORTRAN 77/VP V12L10
      SUBROUTINE GR2D3(CIER)
      IMPLICIT REAL*8 ( A-H,O-Z )
      DATA ISW/1/
      DATA I2*(4000),
      DATA ISW/1/
      DATA 00004800

      C.TN WORK AREA FOR VECTORIALIZATION
      COMMON /VEC2/I2FLG
      DIMENSION I2*(4000),
      DATA ISW/1/
      DATA 00004900

      C
      IF(LISW.EQ.1) THEN
      V-----DO 4010 I1 = 1,N
      V-----I2*(I1) = 0
      V-----CONTINUE
      ISW=0
      C-----END IF
      C
      * MAIN LOOP ALONG ALL PARTICLES
      C-----DO 4000 I1=1,N
      C-----V-----C
      C-----V-----IX=X*(I1,1)+0.55D0
      C-----V-----IY=X*(I1,2)+0.55D0
      C-----V-----IZ=X*(I1,3)+0.55D0
      C-----V-----BOX NUMBER
      C-----V-----I2=1+IX+IALP*1Y+IBET*IZ
      C-----V-----I2*(I1) = 1
      C-----V-----IF(I2.GT.IGAM)
      C-----V-----IF(I2*(I1).NE.I12) THEN
      C-----V-----I2FLG = 1
      C-----V-----I2*(I1) = 12
      C-----V-----END IF
      C-----V-----4000 CONTINUE
      C.TN
      C-----IF(I2FLG.EQ.1) THEN
      *VOCL LOOP,NDVREC
      V-----DO 4100 I1 = 1,N
      V-----I2 = I2*(I1)
      C
      IF(IBOX(6,I2)=I1
      IBOX(6,I2)=I1
      V-----GO TO 800 TEST THE CONTENTS OF BOX
      V-----V-----00005900
      V-----V-----00006000
      V-----V-----00006100
      V-----V-----00006200
      V-----V-----00006300
      V-----V-----00006400
      V-----V-----00006500
      V-----V-----00006600
      V-----V-----00006700
      V-----V-----00006800
      V-----V-----00006900
      V-----V-----00007000
      V-----V-----00007100
      V-----V-----00007200
      V-----V-----00007300
      V-----V-----00007400
      V-----V-----00007500
      C
      IBOX(4,I2)=I1
      GO TO 3000
      V-----1000 CONTINUE
      V-----IF(IBOX(3,I2).NE.0)
      C
      IBOX(3,I2)=I1
      GO TO 3000
      V-----1500 CONTINUE

      SSN=0040
      SSN=0041
      SSN=0042
      SSN=0043
      SSN=0044
      SSN=0045
      SSN=0046

```

Fig.2.8 Modified Subroutine GR2D3

```

★ GR2D3  < 2 >

      FILE=J00001.DDGR2DVP.FORT77(GR2D3)          FORTRAN 77/VP V12L10
C-----*---1-----*---2-----*---3-----*---4-----*---5-----*---6-----*---7-----*---8
      IF(IBOX(2,I2).NE.0)  GO TO 1600           00007600
      IBOX(2,I2)=11          PARTICLE 1,2 ---> THIRD PART 00007700
      V   GO TO 3000          00007800
      V   00007900
      C
      V   1600 CONTINUE          00008000
      V   IF(IBOX(1,I2).NE.0)  GO TO 8100           00008100
      C
      V   IBOX(1,I2)=11          PARTICLE 1,2,3---> FOURTH PART 00008200
      V   00008300
      C
      V   3000 LIBND(I1)=12          00008400
      V   C4000 CONTINUE          00008500
      V   -4100 CONTINUE          00008600
      +-----END IF          00008700
      IER=0          00008700
      00008800
      00008900
      8000 IER=1          00009000
      8000 IER=2          00009100
      9000 CONTINUE          00009200
      9300 CONTINUE          00009300
      RETURN          00010700
      END          00010800
      00010900

```

Fig.2.8 Modified Subroutine GR2D3 (continued)

```

# GR2D41 < 1 >
FILE=J0001.DDGR2D.FORT77(GR2D41)
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
SUBROUTINE GR2D41(I1)
IMPLICIT REAL*8 ( A-H,O-Z )
C
COMMON /COMBIN/KK,IIII(50),III4(50)
SSN=0002
SSN=0003
SSN=0001
C 1. FUNCTION
C   (1) FOR DIAMOND TYPE OF PART-2
C   (2) CALCULATION OF ACCELERATION AND POTENTIAL ENERGY
C 2. INPUT
C     I1      PARTICLE NUMBER (INDICATED)
C     Z21    X-COMPONENT OF ACCELERATION SUMMED UP
C     (4000)
C     Z22    Y-
C     (4000)
C     Z23    Z-
C     (4000)
C     IBND   BOUNDARY TYPE
C     (4000)
IBND   PARTICLE NUMBER --> BOX NUMBER
C     (4000)
C     IAUX  START ENTRY OF THE TABLE OF TABLES IK(-)
C     (27)
IAUX  LENGTH OF THE TABLE OF TABLES IK(-)
C     (27)
C     IK    GENERAL NEIGHBOURS IN A 3X3X3 BOX ABOUT A GENERAL
C     (2735) PARTICLE FOR EACH BOUNDARY TYPE (27)
C     IBOX  27 TABLES ARE COMBINED INTO 1 TABLE
C     (4096) BOXES THAT CONTAIN PARTICLE NUMBERS
C     X     COORDINATE OF PARTICLES
C     (4000,3)
C     E     ACCELERATION ALONG 'R', 'THETA'.
C     (1001,2)
EE     POTENTIAL ON 'R', 'THETA'.
C     (1001,2)
IA2   MAXIMUM VALUE OF X-COORDINATES
IB2   Y-
IC2   Z-
TB    INITIAL VALUE OF COORDINATES
C     (4000,3)
C 3. OUTPUT
C     ENGP0  POTENTIAL ENERGY
C     Z21    X-COMPONENT OF ACCELERATION SUMMED UP
C     (4000)
C     Z22    Y-
C     (4000)
C     Z23    Z-
C     (4000)
C 4. LOCAL VARIABLES
C     NF(4)  PARTICLE NUMBER OF FOUR NEIGHBOURS
C     RNF(4) DISTANCE BETWEEN THE INDICATED PARTICLE AND EACH
C           NEIGHBOUR PARTICLE
C
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
```

Fig.2.9 Subroutine GR2D41

```

★ GR2D41 ( 2 )

      FILE=J0001.DDGR2D.FORT77(GR2D41)          FORTRAN 77/VP V12L10
      C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
      C      5. CALLED BY GR2D40
      C      6. CALLS NOTHING
      C      7. RESTRICTION
      C      8. REFERENCE

      SSN=0004
      C      DIMENSION Z71(4000),Z72(4000),Z73(4000),AC(3),
      C             IBND(4000),IBND(4000),
      C             IK(2735),IAUX(27),IDUX(27),
      C             IBOX(6),IOP(6),
      C             E(1001,3),EE(1001,3),E1(3),E2(3),
      C             X(4000,3),TB(4000,3),
      C             ICOV(4000)

      C      COMMON /GEOM2/GEOM2(43377),
      C             /DYNA/DYNA(42498),
      C             /WORK/WORK(8006)

      C      EQUIVALENCE (GEOM2(10),IALP),
      C             (GEOM2(11),IBET),
      C             (GEOM2(12),IGAM),
      C             (GEOM2(13),TB(1,1)),
      C             (GEOM2(12013),IBND(1)),
      C             (GEOM2(16013),IK(1)),
      C             (GEOM2(18748),IAUX(1)),
      C             (GEOM2(18775),IDUX(1)),
      C             (GEOM2(18802),IBOX(1,1)),
      C             (DYN(11),X(1,1)),
      C             (DYN(24001),AC(1)),
      C             (DYN(24004),Z21(1)),
      C             (DYN(28004),Z22(1)),
      C             (DYN(32004),Z23(1)),
      C             (DYN(36004),E(1,1)),
      C             (DYN(39007),EE(1,1)),
      C             (DYN(42013),E2(1)),
      C             (DYN(42488),ENGPO),
      C             (WORK(7),IBND(1)),
      C             (WORK(4007),ICOV(1))

      SSN=0005
      C      EQUIVALENCE (GEOM2(11),IALP),
      C             (GEOM2(12),IBET),
      C             (GEOM2(13),TB(1,1)),
      C             (GEOM2(12013),IBND(1)),
      C             (GEOM2(16013),IK(1)),
      C             (GEOM2(18748),IAUX(1)),
      C             (GEOM2(18775),IDUX(1)),
      C             (GEOM2(18802),IBOX(1,1)),
      C             (DYN(11),X(1,1)),
      C             (DYN(24001),AC(1)),
      C             (DYN(24004),Z21(1)),
      C             (DYN(28004),Z22(1)),
      C             (DYN(32004),Z23(1)),
      C             (DYN(36004),E(1,1)),
      C             (DYN(39007),EE(1,1)),
      C             (DYN(42013),E2(1)),
      C             (DYN(42488),ENGPO),
      C             (WORK(7),IBND(1)),
      C             (WORK(4007),ICOV(1))

      SSN=0006
      C      EQUIVALENCE (GEOM2(11),IALP),
      C             (GEOM2(12),IBET),
      C             (GEOM2(13),TB(1,1)),
      C             (GEOM2(12013),IBND(1)),
      C             (GEOM2(16013),IK(1)),
      C             (GEOM2(18748),IAUX(1)),
      C             (GEOM2(18775),IDUX(1)),
      C             (GEOM2(18802),IBOX(1,1)),
      C             (DYN(11),X(1,1)),
      C             (DYN(24001),AC(1)),
      C             (DYN(24004),Z21(1)),
      C             (DYN(28004),Z22(1)),
      C             (DYN(32004),Z23(1)),
      C             (DYN(36004),E(1,1)),
      C             (DYN(39007),EE(1,1)),
      C             (DYN(42013),E2(1)),
      C             (DYN(42488),ENGPO),
      C             (WORK(7),IBND(1)),
      C             (WORK(4007),ICOV(1))

      SSN=0007
      C      EQUIVALENCE (GEOM2(11),IALP),
      C             (GEOM2(12),IBET),
      C             (GEOM2(13),TB(1,1)),
      C             (GEOM2(12013),IBND(1)),
      C             (GEOM2(16013),IK(1)),
      C             (GEOM2(18748),IAUX(1)),
      C             (GEOM2(18775),IDUX(1)),
      C             (GEOM2(18802),IBOX(1,1)),
      C             (DYN(11),X(1,1)),
      C             (DYN(24001),AC(1)),
      C             (DYN(24004),Z21(1)),
      C             (DYN(28004),Z22(1)),
      C             (DYN(32004),Z23(1)),
      C             (DYN(36004),E(1,1)),
      C             (DYN(39007),EE(1,1)),
      C             (DYN(42013),E2(1)),
      C             (DYN(42488),ENGPO),
      C             (WORK(7),IBND(1)),
      C             (WORK(4007),ICOV(1))

      SSN=0008
      C      EQUIVALENCE (GEOM2(11),IALP),
      C             (GEOM2(12),IBET),
      C             (GEOM2(13),TB(1,1)),
      C             (GEOM2(12013),IBND(1)),
      C             (GEOM2(16013),IK(1)),
      C             (GEOM2(18748),IAUX(1)),
      C             (GEOM2(18775),IDUX(1)),
      C             (GEOM2(18802),IBOX(1,1)),
      C             (DYN(11),X(1,1)),
      C             (DYN(24001),AC(1)),
      C             (DYN(24004),Z21(1)),
      C             (DYN(28004),Z22(1)),
      C             (DYN(32004),Z23(1)),
      C             (DYN(36004),E(1,1)),
      C             (DYN(39007),EE(1,1)),
      C             (DYN(42013),E2(1)),
      C             (DYN(42488),ENGPO),
      C             (WORK(7),IBND(1)),
      C             (WORK(4007),ICOV(1))

      SSN=0009
      C      DATA ISW/1/
      C             DATA RO/0.8660254DO/
      C             DATA THETA0/1.910633D0/ PREPARATION
      C             IF (ISW.NE.1) GO TO 900
      C             ISW=0
      C             IA=IALP-1
      C             IB=IBET/IALP-1
      C             IC=IGAM/IBET-1
      C             CONTINUE
      900
      SSN=0010
      SSN=0011
      SSN=0012
      SSN=0013
      SSN=0014
      SSN=0015
      SSN=0016
      SSN=0017
      SSN=0018
      SSN=0019
  
```

Fig.2.9 Subroutine GR2D41 (continued)

```

★ GR2D41 ( 3 )

      FILE=J0001.DDGGR2D.FORT77(GR2D41)      FORTRAN 77 / VP V12L10
      *-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*
      C-----1-----2-----3-----4-----5-----6-----7-----8
      C-----V-----V-----V-----V-----V-----V-----V-----V-----V
      SSN=0020          DO 910 I=1,4
      SSN=0021          NF(I)=0
      SSN=0022          RNF(I)=0.D0
      SSN=0023          DO 910 J=1,3
      SSN=0024          V3=-910 R(J,I)=0.D0
      SSN=0025          DO 920 I=1,6
      SSN=0026          V=-920 THETA(I)=0.D0
      +
      C-----ACCELERATION --> AC(-)
      C-----AC(1)=ZZ1(I1)
      C-----AC(2)=ZZ2(I1)
      C-----AC(3)=ZZ3(I1)
      C-----BOUNDARY TYPE --> 10
      C----- (0---26)
      SSN=0027          IO=IBND(I1)
      SSN=0028          IF(IBND(I1),LT.0) IO=0
      SSN=0029          NUMBER OF VALENCE
      SSN=0030          NVAL=4
      SSN=0031          IF(I0.GE.-1.AND.-IO.LE.-6) NVAL=2
      SSN=0032          IF(I0.GE.-7.AND.-IO.LE.-18) NVAL=1
      SSN=0033          IF(I0.EQ.-19) NVAL=1
      SSN=0034          IF(I0.GE.20.AND.I0.LE.22) NVAL=0
      SSN=0035          IF(I0.GE.23.AND.I0.LE.22) NVAL=0
      SSN=0036          IF(I0.EQ.-25) NVAL=0
      SSN=0037          IF(I0.EQ.-26) NVAL=0
      SSN=0038          J1=IBND(I1)
      SSN=0039          IBND(I1)=0
      SSN=0040          C-----STARTING ENTRY ON TABLE OF TABLES00013500
      SSN=0041          C-------> I2                         00013600
      SSN=0042          C-----THE LAST ENTRY ON TABLE OF TABLE00013800
      SSN=0043          S --> IC                         00013900
      SSN=0044          C-----LOOP ALONG I2 / SEARCH BOXES
      SSN=0045          C-----SET UP POSSIBLE NEIGHBOR BOX
      SSN=0046          C-----NUMBER
      SSN=0047          C-----BOX NUMBER OF RELATED PARTICLE
      SSN=0048          C-----ICOMB=0
      SSN=0049          NAL=1
      SSN=0050          DO 10 K=1,KK
      SSN=0051          IF(I1.EQ.-1111(K)) GO TO 11
      SSN=0052          IF(I1.EQ.-1114(K)) GO TO 12
      +
      SSN=0053          10 CONTINUE
      SSN=0054          GO TO 1000
      SSN=0055          11 I4=1114(K)
      SSN=0056          ICOMB=1
      SSN=0057          NAL=2
      C-----FIXED BOND IS STORED IN THE FIRST BOX
      C-----GO TO 2001
      C-----12 I4=1111(K)
      C-----ICOMB=1
      C-----NAL=2
      C-----FIXED BOND IS STORED IN THE FIRST BOX
      C-----GO TO 2001
      +
      C-----V-----V-----V-----V-----V-----V-----V-----V-----V-----V-----V-----V-----V
      00010800
      00010900
      00011000
      00011100
      00011200
      00011300
      00011400
      00011500
      00011600
      00011700
      00011800
      00011900
      00012000
      00012100
      00012200
      00012300
      00012400
      00012500
      00012600
      00012700
      00012800
      00012900
      00013000
      00013100
      00013200
      00013300
      00013400
      00013500
      00013600
      00013700
      00013800
      00013900
      00014000
      00014100
      00014200
      00014300
      00014400
      00014500
      00014600
      00014700
      00014800
      00014900
      00015000
      00015100
      00015200
      00015300
      00015400
      00015500
      00015600
      00015700
      00015800
      00015900
      00016000
      00016100

```

Fig.2.9 Subroutine GR2D41 (continued)

```

★ GR2D41 ( 4 )

FILE=J0001.DDGR2D.FORT77(GR2D41) FORTRAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C-----CONTINUE
      1000  ICOMB=0
      C   I3=J1+1K(I12)          13
      C   IF(I3.LE.0)           GO TO 7000
      C   IF(I3.GE.-497)        GO TO 7000
      C-----DO 1500 I=1,6
      C-----V--1500 IBOXW(I)=IBOX(I,13)
      C-----TEST FOR BOX
      C-----LOOP IN A BOX
      C-----CONTINUE
      C   I4=IBOXW(6)           TAKE UP THE PARTICLE NUMBER
      C-----EMPTY IN BOX ?
      C   IF(I4.EQ.0)           GO TO 7000
      C   IF(ICOMB.EQ.1)        GO TO 2001
      C-----DO 100 K=1,KK
      S-----IF(I1.EQ.III1(K).AND.I4.EQ.III4(K)) GO TO 3000
      S-----IF(I1.EQ.III1(K).AND.I4.EQ.III1(K)) GO TO 3000
      C-----CONTINUE
      C   2001 CONTINUE
      XW=X(I4,1)-X(C1,1)
      YW=X(I4,2)-X(C1,2)
      ZW=X(I4,3)-X(C1,3)
      WRK2=XW**2+YW**2+ZW**2
      WRK=SQRT(WRK2)
      C-----THE SAME OR ALREADY CALCULATED ?
      I11=I1-14
      IF(I11.EQ.0)           GO TO 3000
      IF(I11.GT.0)           GO TO 2400
      PHI3(RP1)-->PHI3(RPN)
      C-----E11=E1(3)*WRK+E2(3)
      E11=E1(3)*WRK+E2(3)
      I5=E11
      H=E11-15
      C-----ACC=E(I5,3)+H*(E(I5+1,3)-E(I5,3))
      ACC=AC(1)+AC(2)+AC(XW/WRK)
      AC(1)=AC(1)+AC(C1)+ACC*XW/WRK
      AC(2)=AC(2)+AC(C2)+ACC*YX/WRK
      AC(3)=AC(3)+AC(C3)+ACC*ZW/WRK
      Z21(I4)=Z21(I4)-ACC*XW/WRK
      Z22(I4)=Z22(I4)-ACC*YW/WRK
      Z23(I4)=Z23(I4)-ACC*ZW/WRK
      C-----ENGPO=ENGPO+EE(I5,3)+H*(EE(I5+1,3)-EE(I5,3))
      ENGPO=ENGPO+EE(I5,3)+H*(EE(I5+1,3)-EE(I5,3))
      C-----REFERENCE CHECK-COUNT
      2400 IF(ICAV(I4).GE.-4)    GO TO 3000
      C-----SEARCH NEIGHBOUR FOUR PARTICLES
      IF(NVAL.EQ.0)           GO TO 3000
      C-----FORMAT(IH,6,6,4,16,E15.7)
      WRITE(6,6300) I1,I4,J1,13,WRK
      C-----FORMAT(IH,6,6,4,16,E15.7)
      C6300 FORMAT(IH,6,6,4,16,E15.7)
      C6400 FORMAT(IH,6,6,4,16,E15.7)

```

Fig. 2.9 Subroutine GR2D41 (continued)

```

★ GR2D41 < 5 >

      FILE=J0001.DDGR20.FORT77(GR2D41)        FORTRAN 77/VP V12L10
      NVAL1=NAL-ICOMB                          5-----*
      DO 2440 I=NVAL1,NVAL                   6-----*
     IF(CRNF(I).EQ.0.0D0.OR.RNF(I).GT.-WRK)   7-----*
     GO TO 2471                                8-----*
      CONTINUE                                     *-----*
      GO TO 3000
      J=NVAL1
      IF(J.EQ.1) GO TO 2473
      RNF(J)=RNF(J-1)
      R(1,J)=R(1,J-1)
      R(2,J)=R(2,J-1)
      R(3,J)=R(3,J-1)
      NF(J)=NF(J-1)
      J=J-1
      GO TO 2472
      RNF(I)=WRK
      R(1,I)=XW/WRK
      R(2,I)=YW/WRK
      R(3,I)=ZW/WRK
      NF(I)=I4
      IF(ICOMB.EQ.1) GO TO 1000
      SHIFT THE PARTICLE NO. IN BOX
      SSN=0109
      SSN=0110
      SSN=0111
      SSN=0112
      SSN=0113
      SSN=0114
      SSN=0115
      SSN=0116
      SSN=0117
      SSN=0118
      SSN=0119
      SSN=0120
      SSN=0121
      SSN=0122
      SSN=0123
      SSN=0124
      SSN=0125
      SSN=0126
      SSN=0127
      SSN=0128
      SSN=0129
      SSN=0130
      3000 CONTINUE
      IBOXW(6)=IBOXW(5)
      IBOXW(5)=IBOXW(4)
      IBOXW(4)=IBOXW(3)
      IBOXW(3)=IBOXW(2)
      IBOXW(2)=IBOXW(1)
      IBOXW(1)=0
      GO TO 2000 OVER ALL NEIGHBOUR BOXES ?
      C
      7000 CONTINUE
      I2=I2+1
      IF(I2.GT.IC)
      GO TO 7500
      C
      7500 CONTINUE
      C
      1-----DO 7520 I=1,4
      IF(NFL(I).NE.0) ICOV(NFL(I))=ICOV(NF(I))+1
      1-----CONTINUE
      C
      7520 CONTINUE
      C
      C
      CALCULATE POTENTIAL AND
      ACCELERATION IN R
      C
      WRITE(6,6000) R
      C6000 FORMAT(1H ,4(3E10.2))
      C
      WRITE(6,6200) NF
      C6200 FORMAT(1H ,4I5)
      C.TN CONVERT TO "FORT77EX" FROM "FORT77HE"
      C.TN ARCS -> ACOS
      1 F(RNF(I)*RNF(2)*NE.O-DO)
      * THETA(I)=ACOS(R(1,1)*R(1,2)+R(2,1)*R(2,2)+R(3,1)*R(3,2))
      C.TN * THETA(I)=ACOS(R(1,1)*R(1,2)+R(2,1)*R(2,2)+R(3,1)*R(3,2),
      C.TN * RNF(I)*RNF(3)*NE.O-DO)
      * THETA(2)=ACOS(R(1,1)*R(1,3)+R(2,1)*R(2,3)+R(3,1)*R(3,3))

SSN=0131
SSN=0132

```

Fig.2.9 Subroutine GR2D41 (continued)

```

★ GR2D41 ( 6 )

C----*---1----*---2----*---3----*---4----*---5----*---6----*---7----*---8----*
C . TN * THETA(2)=ACOS(R(1,1)*R(1,3)+R(2,1)*R(2,3)+R(3,1)*R(3,3)) FORTRAN 77/VP V12L10
SSN=0133 IF(CRNF(1)*RNF(2).NE.0.D0)
* THETA(3)=ACOS(R(1,1)*R(1,4)+R(2,1)*R(3,1)*R(3,4))
C . TN * THETA(3)=ACOS(R(1,1)*R(1,4)+R(2,1)*R(2,4)+R(3,1)*R(3,4))
IF(CRNF(2)*RNF(3).NE.0.D0)
* THETA(4)=ACOS(R(1,2)*R(1,3)+R(2,2)*R(2,3)+R(3,2)*R(3,3))
C . TN * THETA(4)=ACOS(R(1,2)*R(1,3)+R(2,2)*R(2,3)+R(3,2)*R(3,3))
IF(CRNF(3)*RNF(4).NE.0.D0)
* THETA(5)=ACOS(R(1,3)*R(1,4)+R(2,3)*R(2,4)+R(3,3)*R(3,4))
C . TN * THETA(5)=ACOS(R(1,3)*R(1,4)+R(2,3)*R(2,4)+R(3,3)*R(3,4))
IF(CRNF(2)*RNF(4).NE.0.D0)
* THETA(6)=ACOS(R(1,2)*R(1,4)+R(2,2)*R(2,4)+R(3,2)*R(3,4))
C . TN * THETA(6)=ACOS(R(1,2)*R(1,4)+R(2,2)*R(2,4)+R(3,2)*R(3,4))
C
PHI1(R1)----->PHI1(R4)
C-----DO 7700 I=1,4
RNFR=RNF(I)
IF(CRNF(I).EQ.0.D0) RNFR=RO
E11=E1(I)*RNFR +E2(I)
I5=E11
H=E11-I5
C
PHI1(I)=EE(I5,1)+H*(EE(I5+1,1)-EE(I5,1))
DPhi1(I)=E(I5,1)+H*(EE(I5+1,1)-E(I5,1))
C-----7700 CONTINUE
C
PHI2(THETA1)----->PHI2(THETA5)
C-----DO 7800 I=1,6
V-----IF(THETA(I).EQ.0.D0) THETA(I)=THETA0
E11=E1(I)*THETA(I)+E2(I)
I5=E11
H=E11-I5
C
PHI2(I)=EE(I5,2)+H*(EE(I5+1,2)-EE(I5,2))
DPhi1(I)=DPhi1(I)*PHI2(I)*PHI2(I+1)
DPhi1(I+1)=DPhi1(I+1)*PHI2(I+2)*PHI2(I+3)
DPhi1(I+2)=DPhi1(I+2)*PHI2(I+3)*PHI2(I+4)*PHI2(I+6)
DPhi1(I+3)=DPhi1(I+3)*PHI2(I+4)*PHI2(I+5)
DPhi1(I+4)=DPhi1(I+4)*PHI2(I+5)*PHI2(I+6)
C
R(PHI1)/R(R1)--->R(PHI1)/R(R4)
C-----S2 -----DO 8000 I=1,4
M2 DPhi1(I)=DPhi1(I)*R(I,1)
M2 DPhi1(I)=DPhi1(I)*R(I,2)
V2 DPhi1(I)=DPhi1(I)*R(I,3)
C
M2 AC(I)=AC(I)+DPDR(I)
M2 AC(I)=AC(I+2)+DPDR(I)
V2 AC(I)=AC(I+3)+DPDR(I)
C-----V2-8000 CONTINUE
C ACC. DUE TO BIAS OF BOND1
DTB=DPhi1(I)*DPHI2(I)*PHI2(I+1)*PHI2(I+2)*PHI2(I+3)
IF(CRNF(I)*RNF(I).NE.0.D0),
SSN=0165
SSN=0166

```

Fig.2.9 Subroutine GR2D41 (continued)

```

★ GR2D41 ( 7 )

FILE=J0001.DDGR2D.FORT77(GR2D41)      FORTRAN 77 / VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
      *   CALL GR2D4A(DTB,R(1,1),R(1,2),RNF(1),RNF(2),THETA(1),AC)
C
      DTB=PHI1(1)*PHI12(1)*DPHI12(2)*PHI12(3)
      IF(RNF(1)*RNF(3).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,1),R(1,3),RNF(1),RNF(3),THETA(2),AC)
C
      DTB=PHI1(1)*PHI12(1)*DPHI12(2)*DPHI12(3)
      IF(RNF(1)*RNF(6).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,1),R(1,4),RNF(1),RNF(4),THETA(3),AC)
C
      C ACC. DUE TO BIAS OF BOND2
      DTB=PHI1(2)*DPHI12(1)*PHI12(4)*DPHI12(6)
      IF(RNF(2)*RNF(2).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,2),R(1,3),RNF(2),RNF(3),THETA(4),AC)
C
      DTB=PHI1(2)*PHI12(1)*DPHI12(4)*DPHI12(6)
      IF(RNF(2)*RNF(4).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,2),R(1,4),RNF(2),RNF(4),THETA(5),AC)
C
      C ACC. DUE TO BIAS OF BOND3
      DTB=PHI1(3)*DPHI12(2)*PHI12(4)*PHI12(5)
      IF(RNF(1)*RNF(3).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,1),R(1,3),RNF(1),RNF(3),THETA(2),AC)
C
      DTB=PHI1(3)*PHI12(2)*DPHI12(4)*DPHI12(5)
      IF(RNF(2)*RNF(3).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,2),R(1,3),RNF(2),RNF(3),THETA(4),AC)
C
      DTB=PHI1(3)*PHI12(2)*DPHI12(4)*DPHI12(5)
      IF(RNF(3)*RNF(4).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,3),R(1,4),RNF(4),RNF(5),THETA(5),AC)
C
      C ACC. DUE TO BIAS OF BOND4
      DTB=PHI1(4)*DPHI12(3)*PHI12(5)*DPHI12(6)
      IF(RNF(1)*RNF(4).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,1),R(1,4),RNF(1),RNF(4),THETA(3),AC)
C
      DTB=PHI1(4)*PHI12(3)*DPHI12(5)*PHI12(6)
      IF(RNF(3)*RNF(4).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,3),R(1,4),RNF(3),RNF(4),THETA(5),AC)
C
      C
      DTB=PHI1(4)*PHI12(3)*PHI12(5)*DPHI12(6)
      IF(RNF(2)*RNF(4).NE.0.ODO)
      *   CALL GR2D4A(DTB,R(1,2),R(1,4),RNF(2),RNF(4),THETA(6),AC)
C
      C
      1-----DO 8010 I=1,4
      +-----8010 IF(RNF(1).NE.0.ODO) RNF(1)=1.0D0
          ENGP0=ENGPO+  

          * RNF(1)*PHI12(1)*PHI12(2)*PHI12(3)
          POTENTIAL ENERGY(NON IONIC)
          00036500
          00036600
          00036700
          00036800
          00036900
SSN=0187
SSN=0188
SSN=0189
SSN=0190
SSN=0191

```

Fig.2.9 Subroutine GR2D41 (continued)

```

★ GR2D41 < 8 >

      FILE=JO001.DDGR2D.FORT7(GR2D41)          FORTRAN 77/VP V12L10
      *-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      *+RNF(2)*      PHI1(2)*PHI2(1)*PHI1(4)*PHI1(6)    00037000
      *+RNF(3)*      PHI1(3)*PHI1(2)*PHI2(2)*PHI1(5)    00037100
      *+RNF(4)*      PHI1(4)*PHI1(3)*PHI1(5)*PHI1(6)    00037200
      RETURN          00037300
      END             00037400

SSN=0192
SSN=0193

```

Fig.2.9 Subroutine GR2D41 (continued)

```

***** GR2D41 < 1 >
      FILE=J0001.DDGR2DVP.FORT77(GR2D41)
      SUBROUTINE GR2D41(N)          FORTRAN 77/VP V12L10
      IMPLICIT REAL*8 ( A-H,O-Z )
      C.TN VECTORIZATION FOR PARAMETER
      C.TN PARAMETER(CLV=30)
      COMMON /COMBIN/KK,L111(50),L114(50)
      FUNCTION
      C (1) FOR DIAMOND TYPE OF PART-2
      C (2) CALCULATION OF ACCELERATION AND POTENTIAL ENERGY
      C 2. INPUT
      C   11  PARTICLE NUMBER (INDICATED)
      C   221 X-COMPONENT OF ACCELERATION SUMMED UP
      C   (4000)  Y-
      C   222  Z-
      C   (4000)  Z-
      C   223  Z-
      C   (4000)  BND  BOUNDARY TYPE
      C   (4000)  IBND PARTICLE NUMBER --> BOX NUMBER
      C   (4000)  IAUX START ENTRY OF THE TABLE OF TABLES IK(-)
      C   (27)  IDUX LENGTH OF THE TABLE OF TABLES IK(-)
      C   (27)  IK GENERAL NEIGHBOURS IN A 3X3X3 BOX ABOUT A GENERAL
      C   (2735) PARTICLE FOR EACH BOUNDARY TYPE(27)
      C   27 TABLES ARE COMBINED INTO 1 TABLE
      C   IBX BOXES THAT CONTAIN PARTICLE NUMBERS
      C   (4096) X COORDINATE OF PARTICLES
      C   (4000,3) E ACCELERATION ALONG 'R' , 'THETA',
      C   (1001,2) EE POTENTIAL ON 'R' , 'THETA'
      C   (1001,2) IA2 MAXIMUM VALUE OF X-COORDINATES
      C   IB2  IC2 TB INITIAL VALUE OF COORDINATES
      C   (4000,3) 4. LOCAL VARIABLES
      C   ENGP0 POTENTIAL ENERGY
      C   221 X-COMPONENT OF ACCELERATION SUMMED UP
      C   (4000) 222 Y-
      C   (4000) 223 Z-
      C   (4000) 4. LOCAL VARIABLES
      C   NF(4) PARTICLE NUMBER OF FOUR NEIGHBOURS
      C
      SSN=0003
      SSN=0004
      SSN=0001
      SSN=0002

```

Fig.2.10 Modified Subroutine GR2D41

```

★ GR2D41 < 2 >

FILE=JO001.DDGR2DVP.FORT77(GR2D41) FORTRAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
C      RNF(4) DISTANCE BETWEEN THE INDICATED PARTICLE AND EACH
C      NEIGHBOUR PARTICLE
C
C      C. 5. CALLED BY GR2D40
C      C. 6. CALLS NOTHING
C      C. 7. RESTRICTION
C      C. 8. REFERENCE
C
C.TN  DIMENSION ZZ1(4000),ZZ2(4000),ZZ3(4000),AC(3),
C        IBND(4000),IZ2(4000),IZ3(4000),AC(4000,3),
C        IK(2735),IAUX(27),IDUX(27),
C        IBOX(6,4096),
C        E(1001,3),EE(1001,3),E1(3),E2(3),
C        X(4000,3),TB(4000,3),
C        ICOV(4000)

C WORK AREA FOR VECTORIALIZATION
COMMON /VEC1/AC
COMMON /VEC2/IB2FLG
DIMENSION J1*(4000),I2*(4000),ICX(4000),NVAL*(4000)
DIMENSION IBTBL(4000,LV),ITIL(4000,LV),NAL*(4000,LV),
          ICOMX(4000,LV),ICNT(LV)
COMMON /GEOM2/GEOM2(43377),
          /DYNA/DYNA(2498),
          /WORK/WORK(8006)

C EQUIVALENCE (GEOM2(10),IALP),
C           (GEOM2(11),IBET),
C           (GEOM2(12),IGAM),
C           (GEOM2(13),TB(1,1)),
C           (GEOM2(12013),IBND(1,1)),
C           (GEOM2(16013),IK(1,1)),
C           (GEOM2(18748),IAUX(1,1)),
C           (GEOM2(18775),IDUX(1,1)),
C           (GEOM2(18802),IBOX(1,1)),
C           (DYNA(1),X(1,1)),
C           (DYNA(24004),ZZ1(1)),
C           (DYNA(28004),ZZ2(1)),
C           (DYNA(32004),ZZ3(1)),
C           (DYNA(36004),E(1,1)),
C           (DYNA(39007),EE(1,1)),(DYNA(42010),E1(1,1)),
C           (DYNA(42013),E2(1)),(DYNA(42488),ENGPO),
C
C.TN  EQUIVALENCE (DYNA(24001),AC(1)),
C           (WORK(7),IBND(1,1)),
C           (WORK(4007),ICOV(1,1))

C.TN  DIMENSION NF(4),RNF(4),THETA(6),R(3,4),IBOXW(6),
C           NF(4000,4),RNF(4000,4),THETA(4000,6),
C           R(4000,3,4)
C.TN  DIMENSION PHI1(4),DPHI1(4),
C           PHI2(6),DPHI2(6),
C           DPDR(3)

SSN=0005
SSN=0006
SSN=0007
SSN=0008
SSN=0009
SSN=0010
SSN=0011
SSN=0012
SSN=0013
SSN=0014

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 < 3 >

C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      FILE=J0001.DDGR2DVP.FORT77(GR2D41) FORTRAN 77/VP V12L10
      DATA ISW/1/
      DATA RO/0.8660254DO/
      DATA THETAO/1.910633DO/
      DATA DO 5000 11 = 1, N
      C-N IF(CISW.NE.-1)
      C-N ISW=0
      C-N IA=IALP-1
      C-N IB=IBET/IALP-1
      C-TN IC=IGAM/IBET-1
      V 900 CONTINUE
      C-TN DO 910 I=1,4
      C-TN NF(C)=0
      C-TN RNFC(I)=0.D0
      C-TN DU 910 J=1,3
      C-910 RC(J,I)=0.D0
      C-TN DO 920 I=1,6
      C.920 THETA(C,I)=0.D0
      C-TN NF(C1,1)=0
      C-TN NF(C1,2)=0
      C-TN NF(C1,3)=0
      C-TN NF(C1,4)=0
      RNE(C1,1)=0.D0
      RNE(C1,2)=0.D0
      RNE(C1,3)=0.D0
      RNE(C1,4)=0.D0
      R(C1,1,1)=0.D0
      R(C1,1,2)=0.D0
      R(C1,1,3)=0.D0
      R(C1,1,4)=0.D0
      R(C1,2,1)=0.D0
      R(C1,2,2)=0.D0
      R(C1,2,3)=0.D0
      R(C1,2,4)=0.D0
      R(C1,3,1)=0.D0
      R(C1,3,2)=0.D0
      R(C1,3,3)=0.D0
      R(C1,3,4)=0.D0
      R(C1,4,1)=0.D0
      R(C1,4,2)=0.D0
      R(C1,4,3)=0.D0
      R(C1,4,4)=0.D0
      THETAC(1,1)=0.D0
      THETAC(1,2)=0.D0
      THETAC(1,3)=0.D0
      THETAC(1,4)=0.D0
      THETAC(1,5)=0.D0
      THETAC(1,6)=0.D0
      C ACCELERATION --> ACC(-)
      C-TN AC(1)=Z21(C11)
      C-TN AC(2)=Z22(C11)
      C-TN AC(3)=Z23(C11)
      V V V 5000 CONTINUE
      C-N SSN=0015
      C-N SSN=0016
      C-N SSN=0017
      C-N SSN=0018
      C-N SSN=0019
      C-N SSN=0020
      C-N SSN=0021
      C-N SSN=0022
      C-N SSN=0023
      C-N SSN=0024
      C-N SSN=0025
      C-N SSN=0026
      C-N SSN=0027
      C-N SSN=0028
      C-N SSN=0029
      C-N SSN=0030
      C-N SSN=0031
      C-N SSN=0032
      C-N SSN=0033
      C-N SSN=0034
      C-N SSN=0035
      C-N SSN=0036
      C-N SSN=0037
      C-N SSN=0038
      C-N SSN=0039
      C-N SSN=0040
      C-N SSN=0041
      C-N SSN=0042
      C-N SSN=0043
      C-N SSN=0044
      C-N SSN=0045
      C-N SSN=0046
      C-N SSN=0047
      C-N SSN=0048
      C-N SSN=0049
  
```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 < 4 >

C----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
FILE=J0001.DDGR2DV.P FORTRAN 77/VP V12L10
C.TN
C----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
C.TN
1 2 ----- IF (ISW.EQ.1) THEN
1 2 ----- DO 5010 II = 1, N
      C
      C
      V
      IO=IBND(II)
      IF (IBND(II).LT.0) IO=0
      C.T
      NVAL=4
      IF (IO.GE.-1.AND.IO.LE.-6) NVAL=2
      C.T
      IF (IO.GE.-7.AND.IO.LE.-18) NVAL=1
      C.T
      IF (IO.EQ.-19) NVAL=1
      C.T
      IF (IO.GE.-20.AND.IO.LE.-22) NVAL=0
      C.T
      IF (IO.GE.-23.AND.IO.LE.-25) NVAL=1
      C.T
      IF (IO.EQ.-26) NVAL=0
      C
      NVAL*(II)=4
      IF (IO.GE.-1.AND.IO.LE.-6) NVAL*(II)=2
      IF (IO.GE.-7.AND.IO.LE.-18) NVAL*(II)=1
      IF (IO.EQ.-19) NVAL*(II)=1
      IF (IO.GE.-20.AND.IO.LE.-22) NVAL*(II)=0
      IF (IO.GE.-23.AND.IO.LE.-25) NVAL*(II)=1
      IF (IO.EQ.-26) NVAL*(II)=0
      C
      STRTING ENTRY ON TABLE OF TABLES00013500
      --> I2
      C.TN
      I2=IAUX(10+1)
      V
      C.TN
      I2*(II)=IAUX(10+1)
      C.TN
      IC=IAUX(10+1)+IDUX((10+1)-1
      V
      IC*(II)=IAUX(10+1)+IDUX((10+1)-1
      V--5010 CONTINUE
      ISW = 0
      +---END IF
      C.TN
      1-----IF (I2FLG.EQ.1) THEN
      1-----C.TN
      1-----MAKE 'II','IBOX','NAL' AND 'ICOMB' LIST FOR VECTORIALIZATION
      1-----C.TN
      1-----II -> II, IB0X -> IBTBL, NAL -> NALY, ICOMB -> ICOMY
      1-----C.TN
      1-----V-----DO 5020 II = 1, LV
      1-----V-----ICNT(II) = 0
      1-----V-----5020 CONTINUE
      1-----C.TN
      1-----V-----DO 5030 II = 1, N
      1-----C
      1-----C.TN
      1-----J1=IBND(II)
      1-----J1*(II)=IBND(II)
      1-----V-----IBND(II)=0
      1-----V-----5030 CONTINUE
      1-----C.TN
      1-----I2MAX = 0
      C.TN
      1-----V-----DO 5100 II = 1, N
      1-----V-----DO 10 K = 1, KK
      1-----V-----DO 3-----
```

Fig.2.10 Modified Subroutine GR2D41 (continued)


```

★ GR2D41 ( 6 )
FILE=J0001.DDGR2DVP.FORT77(GR2D41)  FORT77/VF V12L10
C-----1-----2-----3-----4-----5-----6-----7-----8
      1 2 3 4
      1 2 3 +-
      1 2 3 4
      1 2 3 +-
      1 2 3 +-
      1 2 +-----30   CONTINUE
      1 2 +-----END IF
      1 2 +-----30   CONTINUE
      1 2 +-----OVER ALL NEIGHBOUR BOXES ?
      1 2 +-----00024500
      1 2 +-----5200 CONTINUE
      1 2 +-----END IF
      1 2 +-----S-----DO 5300 II = 1, 12MAX
      1 2 +-----*VOC1 LOOP NOVREC
      1 2 +-----DO 5400 IL = 1, ICNT(II)
      1 2 +-----IL = IL(IL,II)
      1 2 +-----I4 = IBTBL(IL,II)
      1 2 +-----C.TN IF(ICOMB.EQ.1) GO TO 2001
      1 2 +-----C IF(ICOM*(IL).EQ.1) GO TO 2001
      1 2 +-----DO 100 K=1,KK
      1 2 +-----C IF(I1-EQ.I111(K).AND.I4-EQ.I114(K)) GO TO 3000
      1 2 +-----C IF(I1-EQ.I114(K).AND.I4-EQ.I11(K)) GO TO 3000
      1 2 +-----C 100 CONTINUE
      1 2 +-----C
      1 2 +-----V 2001 CONTINUE
      1 2 +-----V XW=X(14,1)-X(11,1)
      1 2 +-----V YW=X(14,2)-X(11,2)
      1 2 +-----V ZW=X(14,3)-X(11,3)
      1 2 +-----V WRK2=XW**2+YW**2+ZW**2
      1 2 +-----V WRK=SQRT(WRK2)
      1 2 +-----C THE SAME OR ALREADY CALCULATED ? 00019000
      1 2 +-----C
      1 2 +-----V I11=I1-I4
      1 2 +-----V C.TN IF(I11-EQ.0) GO TO 3000
      1 2 +-----V IF(I11-EQ.0) GO TO 5400
      1 2 +-----V IF(I11.GT.0) PHI3(RP1)--->PHI3(RPN)
      1 2 +-----V E11=E1(3)*WRK+E2(3)
      1 2 +-----V I5=E11
      1 2 +-----V H=E11-15
      1 2 +-----C
      1 2 +-----V ACC=E(15,3)+H*(E(15+1,3)-E(15,3))
      1 2 +-----C.TN AC(1)=AC(1)+ACC*XW/WRK
      1 2 +-----C.TN AC(2)=AC(2)+ACC*YW/WRK
      1 2 +-----C.TN AC(3)=AC(3)+ACC*ZW/WRK
      1 2 +-----V AC(I1,1)=AC(I1,1)+ACC*XW/WRK
      1 2 +-----V AC(I1,2)=AC(I1,2)+ACC*YW/WRK
      1 2 +-----V AC(I1,3)=AC(I1,3)+ACC*ZW/WRK
      1 2 +-----V Z21(I4)=Z21(I4)-ACC*XW/WRK
      1 2 +-----V Z22(I4)=Z22(I4)-ACC*YW/WRK
      1 2 +-----V Z23(I4)=Z23(I4)-ACC*ZW/WRK
      1 2 +-----C
      1 2 +-----V ENGPU=ENGPU+EE(15,3)+H*(EE(15+1,3)-EE(15,3))
      1 2 +-----C 2400 IF(ICOV(I4).GE.4) GO TO 3000
      1 2 +-----REFERENCE CHECK,COUNT
      1 2 +-----00020700
      1 2 +-----00020800
      1 2 +-----00020900

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

★ GR2D41 (7)

```

FILE=J0001.DDGR2DVP.FORT77(GR2D41)  FORTRAN 77/VP V12L10
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C 2400 IF(CICOV(I4).GE.4)      GO TO 5400   SEARCH NEIGHBOUR FOUR PARTICLES
C  TN  IF(CNVAL.EQ.0)           GO TO 3000
V  C  IF(CNVAL*(I1).EQ.0)       GO TO 5400
V  C  WRITE(6,6300) I1,I4,J1,13,WRK
C  WRITE(6,6400) RNF
C 6300 FORMAT(IH ,4I6,E15.7)
C6400 FORMAT(IH ,4E20.7)
C  TN  NVAL1=NAL-ICOMB
C  TN  NVAL1=NAL*ICOM*(I1)
C  TN  DO 2440 I=NVAL1,NVAL
C  TN  DO 2440 I=NVAL1,NVAL*(I1)
C  TN  IF(CRNF(I1).EQ.0.ODO.OR.RNF(I1).GT.WRK) GO TO 2471
C  TN  IF(CRNF(I1,I).EQ.0.ODO.OR.RNF(I1,I).GT.WRK) GO TO 2471
C  TN  IF(CRNF(I1,I).EQ.0.ODO.OR.RNF(I1,I).GT.WRK) GO TO 2471
C2440 CONTINUE
C  TN  I = 1
V  C  TN  NVAL1 = 1
V  C  TN  IF(CNAL*(I1,I1).EQ.2.AND.ICOM*(I1,I1).EQ.1.OR.
V  1  NAL*(I1,I1).EQ.1.AND.ICOM*(I1,I1).EQ.0) THEN
V  1  IF(CNAL*(I1).EQ.1) THEN
V  1  I = 1
V  1  IF(CRNF(I1,I1).EQ.0.ODO.OR.RNF(I1,I1).GT.WRK) GO TO 2471
V  1  ELSE IF(CNAL*(I1).EQ.2) THEN
V  1  I = 1
V  1  IF(CRNF(I1,I1).EQ.0.ODO.OR.RNF(I1,I1).GT.WRK) THEN
V  1  RNF(I1,I2)=RNF(I1,I1)
V  1  R(I1,I2)=R(I1,I1)
V  1  R(I1,I3)=R(I1,I1)
V  1  NF(I1,I2)=NF(I1,I1)
V  1  GO TO 2471
END IF
I = 2
IF(CRNF(I1,I2).EQ.0.ODO.OR.RNF(I1,I2).GT.WRK) GO TO 2471
ELSE IF(CNAL*(I1).EQ.3) THEN
I = 1
IF(CRNF(I1,I1).EQ.0.ODO.OR.RNF(I1,I1).GT.WRK) THEN
RNF(I1,I3)=RNF(I1,I2)
R(I1,I3)=R(I1,I2)
NF(I1,I3)=NF(I1,I2)
RNF(I1,I2)=RNF(I1,I1)
R(I1,I2)=R(I1,I1)
R(I1,I3)=R(I1,I1)
NF(I1,I2)=NF(I1,I1)
GO TO 2471
END IF
IF(CRNF(I1,I2).EQ.0.ODO.OR.RNF(I1,I2).GT.WRK) THEN
I = 1
RNF(I1,I3)=RNF(I1,I2)
R(I1,I3)=R(I1,I2)
R(I1,I2)=R(I1,I1)
R(I1,I3)=R(I1,I1)
R(I1,I2)=R(I1,I1)
NF(I1,I3)=NF(I1,I1)
GO TO 2471
END IF
IF(CRNF(I1,I3).EQ.0.ODO.OR.RNF(I1,I3).GT.WRK) THEN
I = 1
RNF(I1,I2)=RNF(I1,I3)
R(I1,I2)=R(I1,I3)
R(I1,I3)=R(I1,I2)
NF(I1,I2)=NF(I1,I3)
GO TO 2471
END IF
IF(CRNF(I1,I3).EQ.0.ODO.OR.RNF(I1,I3).GT.WRK) THEN
I = 1
RNF(I1,I2)=RNF(I1,I3)
R(I1,I2)=R(I1,I3)
R(I1,I3)=R(I1,I2)
NF(I1,I2)=NF(I1,I3)
GO TO 2471
END IF

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 ( 8 )
      FILE=J0001.DDGR2DVP.FORT77(GR2D41)  FORTRAN 77/VP V12L10
      C-----1-----2-----*-----3-----4-----*-----5-----6-----*-----7-----8
      V      V      V      V      V      V      V      V      V      V      V      V
      X = 2
      NF(I1,3) = NFN(I1,2)
      GO TO 2471
      END IF
      X = 3
      IF(RNF(I1,3).EQ.0.0D0.OR.RNF(I1,3).GT.WRK) GO TO 2471  00021800
      ELSE IF(NVAL*(I1).EQ.4) THEN
      IF(RNF(I1,1).EQ.0.0D0.OR.RNF(I1,1).GT.WRK) THEN
      RNF(I1,4) = RNF(I1,3)
      RCI1,1,4) = RCI1,1,3)
      RCI1,2,4) = RCI1,2,3)
      RCI1,3,4) = RCI1,3,3)
      RCI1,4,4) = RCI1,4,3)
      RCI1,5,4) = RCI1,5,3)
      NF(I1,4) = NF(I1,3)
      RNF(I1,4) = RNF(I1,2)
      RII1,1,3) = RII1,1,2)
      RII1,2,3) = RII1,2,2)
      RII1,3,3) = RII1,3,2)
      RII1,4,3) = RII1,4,2)
      RII1,5,3) = RII1,5,2)
      NF(I1,2) = RNF(I1,1)
      RNF(I1,1) = RNF(I1,1)
      RCI1,1,2) = RCI1,1,1)
      RCI1,2,2) = RCI1,2,1)
      RCI1,3,2) = RCI1,3,1)
      RNF(I1,2) = RNF(I1,1)
      I = 1
      GO TO 2471
      END IF
      IF(RNF(I1,2).EQ.0.0D0.OR.RNF(I1,2).GT.WRK) THEN
      RNF(I1,4) = RNF(I1,3)
      RCI1,1,4) = RCI1,1,3)
      RCI1,2,4) = RCI1,2,3)
      RCI1,3,4) = RCI1,3,3)
      RNF(I1,4) = RNF(I1,3)
      RNF(I1,3) = RNF(I1,2)
      RNF(I1,2) = RNF(I1,1)
      RCI1,1,3) = RCI1,1,2)
      RCI1,2,3) = RCI1,2,2)
      RNF(I1,1) = RNF(I1,2)
      RNF(I1,2) = RNF(I1,3)
      RNF(I1,3) = RNF(I1,4)
      NF(I1,2) = NF(I1,1)
      I = 2
      GO TO 2471
      END IF
      IF(RNF(I1,3).EQ.0.0D0.OR.RNF(I1,3).GT.WRK) THEN
      RNF(I1,4) = RNF(I1,3)
      RCI1,1,4) = RCI1,1,3)
      RCI1,2,4) = RCI1,2,3)
      RCI1,3,4) = RCI1,3,3)
      NF(I1,4) = NF(I1,3)
      I = 3
      GO TO 2471
      END IF
      I = 4
      IF(RNF(I1,4).EQ.0.0D0.OR.RNF(I1,4).GT.WRK) GO TO 2471  00021800
      ELSE
      C,TN
      NVAL1 = 2
      END IF

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 ( 9 )

FILE=J0001.DGGR2DVP.FORT77(GR2D41) FORTRAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
      1 2 3
      1 2 3 4      V = 2
      1 2 3 4      V = 2
      1 2 3 4      IF(NVAL*(IL,1),EQ.-2.AND.ICOMM*(IL,11),EQ.0) THEN
      1 2 3 4      V = 2
      1 2 3 4      IF(NVAL*(IL,2),EQ.-2) THEN
      1 2 3 4      IF(RNF(IL,1,2).EQ.0.ODO.OR.RNF(IL,2).GT.WRK) GO TO 2471
      1 2 3 4      00021800
      1 2 3 4      ELSE IF(NVAL*(IL,1).EQ.3) THEN
      1 2 3 4      IF(RNF(IL,1,2).EQ.0.ODO.OR.RNF(IL,1,2).GT.WRK) THEN
      1 2 3 4      RNF(IL,1,3) = RNF(IL,1,2)
      1 2 3 4      R(C1,1,1,3) = R(C1,1,1,2)
      1 2 3 4      R(C1,1,2,3) = R(C1,1,2,2)
      1 2 3 4      R(C1,1,3,3) = R(C1,1,3,2)
      1 2 3 4      NF(C1,1,3) = NF(C1,1,2)
      1 2 3 4      GO TO 2471
      1 2 3 4      00022400
      1 2 3 4      END IF
      1 2 3 4      V = 3
      1 2 3 4      IF(RNF(IL,1,3).EQ.0.ODO.OR.RNF(IL,1,3).GT.WRK) GO TO 2471
      1 2 3 4      00021800
      1 2 3 4      ELSE IF(NVAL*(IL,1).EQ.4) THEN
      1 2 3 4      V = 2
      1 2 3 4      IF(NVAL*(IL,2).EQ.0.ODO.OR.RNF(IL,1,2).GT.WRK) THEN
      1 2 3 4      RNF(IL,1,4) = RNF(IL,1,3)
      1 2 3 4      R(C1,1,1,4) = R(C1,1,1,3)
      1 2 3 4      R(C1,1,2,4) = R(C1,1,2,3)
      1 2 3 4      R(C1,1,3,4) = R(C1,1,3,3)
      1 2 3 4      RNF(IL,1,4) = NF(C1,1,3)
      1 2 3 4      RNF(IL,1,3) = RNF(IL,1,2)
      1 2 3 4      R(C1,1,1,3) = R(C1,1,1,2)
      1 2 3 4      R(C1,1,2,3) = R(C1,1,2,2)
      1 2 3 4      R(C1,1,3,3) = R(C1,1,3,2)
      1 2 3 4      NF(C1,1,3) = NF(C1,1,2)
      1 2 3 4      I = 2
      1 2 3 4      GO TO 2471
      1 2 3 4      END IF
      1 2 3 4      IF(RNF(IL,1,3).EQ.0.ODO.OR.RNF(IL,1,3).GT.WRK) THEN
      1 2 3 4      RNF(IL,1,4) = RNF(IL,1,3)
      1 2 3 4      P(C1,1,1,4) = R(C1,1,1,3)
      1 2 3 4      R(C1,1,2,4) = R(C1,1,2,3)
      1 2 3 4      R(C1,1,3,4) = R(C1,1,3,3)
      1 2 3 4      NF(C1,1,4) = NF(C1,1,3)
      1 2 3 4      I = 3
      1 2 3 4      GO TO 2471
      1 2 3 4      END IF
      1 2 3 4      V = 4
      1 2 3 4      IF(RNF(IL,1,4).EQ.0.ODO.OR.RNF(IL,1,4).GT.WRK) GO TO 2471
      1 2 3 4      00021800
      1 2 3 4      ELSE IF(NVAL*(IL,1).EQ.5) THEN
      1 2 3 4      V = 2
      1 2 3 4      C.TN GO TO 3000
      1 2 3 4      V = 2471 RNF(IL,1)=WRK
      1 2 3 4      C.TN R(1,1)=XW/WRK
      1 2 3 4      C.TN R(2,1)=YW/WRK
      1 2 3 4      C.TN R(3,1)=ZW/WRK
      1 2 3 4      R(C1,1,1,1)=XW/WRK
      1 2 3 4      R(C1,1,2,1)=YW/WRK
      1 2 3 4      R(C1,1,3,1)=ZW/WRK
      1 2 3 4      NF(C1,1)=14
      1 2 3 4      00022000
      1 2 3 4      00022000
      1 2 3 4      00023000
      1 2 3 4      00023100
      1 2 3 4      00023200
      1 2 3 4      00023300
      1 2 3 4      00023400
      1 2 3 4      00023100
      1 2 3 4      00023200
      1 2 3 4      00023300
      1 2 3 4      00023400

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 < 10 >

      FILE=J0001.DDGR2DV.P.FORT77(GR2D41)  FORTRAN 77/VP V12L10
      C----*----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
      1 2          NF(1,1)=14
      1 +          V
      1 +          SSN=0300 CONTINUE
      1 +          SSN=0301 CONTINUE
      C-TN          *VOCAL LOOP, NOVREC
      V          --DO 5500 11 = 1, N
      C          00025200
      C-TN          DO 7520 I=1,4
      C-TN          IF(NF(I1).NE.0) ICOV(NF(I1))=ICOV(NF(I1))+1
      C7520          CONTINUE
      V          00025300
      V          IF(NF(C1,1).NE.0) ICOV(NF(C1,1))=ICOV(NF(C1,1))+1
      V          IF(NF(C1,2).NE.0) ICOV(NF(C1,2))=ICOV(NF(C1,2))+1
      V          IF(NF(C1,3).NE.0) ICOV(NF(C1,3))=ICOV(NF(C1,3))+1
      V          IF(NF(C1,4).NE.0) ICOV(NF(C1,4))=ICOV(NF(C1,4))+1
      C-TN          CALCULATE POTENTIAL AND
      C-TN          CONVERT TO "FORTRAN" FROM "FORTRAN"
      C-TN          ARCS -> ACOS
      C-TN          IF(CRNF(1)*RNF(2).NE.0.D0)
      C-TN          * THETA(1)=ARCS(RC1,1)*R(1,2)+R(2,1)*R(3,1)*R(3,2)
      V          00025400
      V          * THETA(1,1)=ARCS(RC1,1)*RNF(1,1)*R(1,2)+R(2,2)+R(3,1)*R(3,2)
      V          * THETA(1,1)=ACOS(RC1,1)*R(1,1,1)*R(1,1,2)+R(1,2,2)+R(1,1,3)*R(1,1,3,2)
      V          00025500
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,2)+R(1,2,2)+R(1,1,3)*R(1,1,3,2)
      C-TN          IF(CRNF(1)*RNF(3).NE.0.D0)
      C-TN          * THETA(2)=ARCS(RC1,2)*R(1,3)+R(2,1)*R(2,3)+R(3,1)*R(3,3)
      V          00025600
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,3)+R(1,2,2)+R(1,1,2,3)+R(1,1,2,3,2)
      V          00025700
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,2)+R(1,2,2)+R(1,1,3)*R(1,1,3,2)
      C-TN          IF(CRNF(1)*RNF(4).NE.0.D0)
      C-TN          * THETA(3)=ARCS(RC1,1)*R(1,4)+R(2,1)*R(2,4)+R(3,1)*R(3,4)
      V          00025800
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,4)+R(1,2,2)*R(1,1,2,4)+R(1,1,2,4,2)
      V          00025900
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,3)+R(1,2,2)*R(1,1,2,4)+R(1,1,2,4,2)
      C-TN          IF(CRNF(2)*RNF(3).NE.0.D0)
      C-TN          * THETA(4)=ARCS(RC1,2)*R(1,3)+R(2,2)*R(2,3)+R(3,2)*R(3,3)
      V          00026000
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,3)+R(1,2,2)*R(1,1,2,4)+R(1,1,2,4,2)
      C-TN          IF(CRNF(3)*RNF(4).NE.0.D0)
      C-TN          * THETA(5)=ARCS(RC1,3)*R(1,4)+R(2,3)*R(2,4)+R(3,3)*R(3,4)
      V          00026100
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,4)+R(1,2,2)*R(1,1,2,4)+R(1,1,2,4,2)
      V          00026200
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,3,2)+R(1,2,2)*R(1,1,2,4)+R(1,1,2,4,2)
      C-TN          IF(CRNF(2)*RNF(5).NE.0.D0)
      C-TN          * THETA(6)=ARCS(RC1,2)*R(1,4)+R(2,2)*R(2,4)+R(3,2)*R(3,4)
      V          00026300
      V          * THETA(1,1)=ACOS(RC1,1)*RNF(1,1)*R(1,1,1)*R(1,1,4)+R(1,2,2)*R(1,1,2,4)+R(1,1,2,4,2)
      C-TN          DO 7700 I=1,4
      C-TN          RNFR=RNF(1)
      C-TN          IF(CRNF(1).EQ.0.D0) RNFR=0
      C-TN          E11=E1(1)*RNFR + E2(1)
      C-TN          PHI1(R1) --->PHI1(R4)
      C-TN          00027500
      C-TN          00027600
      C-TN          00027700
      C-TN          00027800
      C-TN          00027900
      C-TN          00028000

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 ( 11 )

FILE=J0001.DDGR2DVP.FORT77(GR2D41)           FORTRAN 77 / VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C   TN   IS=E11
C   TN   H=E11-15
C
C   TN   PHI1(I)=EE(I5,1)+H*(EE(I5+1,1)-EE(I5,1))
C   TN   DPHI1(I)=E(I5,1)+H*(E(I5+1,1)-E(I5,1))
C7700  CONTINUE
C7700  IF(RNFR=RNFC(1,1))
C7700    IF(RNFC(I1,1).EQ.0.D0) RNFR=RO
C7700    E11=E1(1)*RNFR +E2(1)
C7700
C7700  IS=E11
C7700  H=E11-15
C7700  PHI1(I)=EE(I5,1)+H*(EE(I5+1,1)-EE(I5,1))
C7700  DPHI1(I)=E(I5,1)+H*(E(I5+1,1)-E(I5,1))
C7700  RNFR=RNFC(1,1)
C7700  IF(RNFC(I1,2).EQ.0.D0) RNFR=RO
C7700    E11=E1(1)*RNFR +E2(1)
C7700
C7700  IS=E11-15
C7700  H=E11-15
C7700  PHI1(2)=EE(I5,1)+H*(EE(I5+1,1)-EE(I5,1))
C7700  DPHI1(2)=E(I5,1)+H*(E(I5+1,1)-E(I5,1))
C7700  RNFR=RNFC(1,3)
C7700  IF(RNFC(I1,3).EQ.0.D0) RNFR=RO
C7700    E11=E1(1)*RNFR +E2(1)
C7700
C7700  IS=E11-15
C7700  H=E11-15
C7700  PHI1(3)=EE(I5,1)+H*(EE(I5+1,1)-EE(I5,1))
C7700  DPHI1(3)=E(I5,1)+H*(E(I5+1,1)-E(I5,1))
C7700  RNFR=RNFC(1,4)
C7700  IF(RNFC(I1,4).EQ.0.D0) RNFR=RO
C7700    E11=E1(1)*RNFR +E2(1)
C7700
C7700  IS=E11-15
C7700  H=E11-15
C7700  PHI1(4)=EE(I5,1)+H*(EE(I5+1,1)-EE(I5,1))
C7700  DPHI1(4)=E(I5,1)+H*(E(I5+1,1)-E(I5,1))
C7700
C7700  DO 7800 I=1,6
C7700    C   TN   IF(<THETAC(I).EQ.0.D0) THETAC(I)=THETAO
C7700    C   TN   E11=E1(2)*THETAC(I)+E2(2)
C7700    C   TN   IS=E11
C7700    C   TN   H=E11-15
C7700
C7700  C   TN   PHI2(I)=EE(I5,2)+H*(EE(I5+1,2)-EE(I5,2))
C7700  C   TN   DPHI2(I)=E(I5,2)+H*(E(I5+1,2)-E(I5,2))
C7700  C   TN   E11=E1(2)*THETAC(I)+E2(2)
C7700    C   TN   IS=E11
C7700    C   TN   H=E11-15
C7700
C7700  C   TN   PHI2(2)=EE(I5,2)+H*(EE(I5+1,2)-EE(I5,2))
C7700  C   TN   DPHI2(2)=E(I5,2)+H*(E(I5+1,2)-E(I5,2))
C7700
C7700  C   TN   SSN=0341
C7700  C   TN   SSN=0342
C7700  C   TN   SSN=0343
C7700  C   TN   SSN=0344
C7700  C   TN   SSN=0345
C7700  C   TN   SSN=0346
C7700  C   TN   SSN=0347
C7700  C   TN   SSN=0348
C7700  C   TN   SSN=0349
C7700  C   TN   SSN=0350

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 ( 12 )

C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
V     FILE=J0001.DDGR2DVP.FORT77(GR2D41)          FORTRAN 77 / VP V12L10
      PH122=EE(15/2)+H*(EE(15+1/2)-EE(15/2))           00029500
      DPH122=E(15/2)+H*(E(15+1/2)-E(15/2))           00029600
      IF(THETA(I1,3)=EQ_0,DO THETA(I1,3)=THETA0
      E11=E1(2)*THETA(I1,3)+E2(2)
      I5=E11
      H=E11-15
      PH124=EE(15/2)+H*(EE(15+1/2)-EE(15/2))
      DPH124=E(15/2)+H*(E(15+1/2)-E(15/2))
      IF(THETA(I1,4)=EQ_0,DO THETA(I1,4)=THETA0
      E11=E1(2)*THETA(I1,4)+E2(2)
      I5=E11
      H=E11-15
      PH125=EE(15/2)+H*(EE(15+1/2)-EE(15/2))
      DPH125=E(15/2)+H*(E(15+1/2)-E(15/2))
      IF(THETA(I1,5)=EQ_0,DO THETA(I1,5)=THETA0
      E11=E1(2)*THETA(I1,5)+E2(2)
      I5=E11
      H=E11-15
      PH126=EE(15/2)+H*(EE(15+1/2)-EE(15/2))
      DPH126=E(15/2)+H*(E(15+1/2)-E(15/2))
      C.TN DPHI1(1)=DPHI1(1)*PHI12(1)*PHI12(2)*PHI12(3)
      C.TN DPHI1(2)=DPHI1(2)*PHI12(1)*PHI12(4)*PHI12(6)
      C.TN DPHI1(3)=DPHI1(3)*PHI12(2)*PHI12(4)*PHI12(5)
      C.TN DPHI1(4)=DPHI1(4)*PHI12(3)*PHI12(5)*PHI12(6)
      DPHI11=DPHI11*PHI121*PHI122*PHI123
      DPHI12=DPHI12*PHI124*PHI125
      DPHI13=DPHI13*PHI122*PHI124*PHI126
      DPHI14=DPHI14*PHI123*PHI125*PHI127
      C.C DO 8000 I=1,4
      C.TN DPDR(1)=DPHI1(1)*R(1,1)
      C.TN DPDR(2)=DPHI1(1)*R(2,1)
      C.TN DPDR(3)=DPHI1(1)*R(3,1)
      C.C AC(1)=AC(1)+DPDR(1)
      C.TN AC(2)=AC(2)+DPDR(2)
      C.TN AC(3)=AC(3)+DPDR(3)
      C8000 CONTINUE
      V>>> DPDR1=DPHI11*R(1,1,1)
      DPDR2=DPHI11*R(1,1,2,1)
      DPDR3=DPHI11*R(1,1,3,1)
      AC(1,1)=AC(1,1)+DPDR1
      AC(1,2)=AC(1,1,2)+DPDR2
      AC(1,3)=AC(1,1,3)+DPDR3
      AC(1,4)=AC(1,1,4)+DPDR4
      DPDR1=DPHI12*R(1,1,1,2)
      DPDR2=DPHI12*R(1,1,2,2)
      SSN=0351 1
      SSN=0352 1
      SSN=0353 1
      SSN=0354 1
      SSN=0355 1
      SSN=0356 1
      SSN=0357 1
      SSN=0358 1
      SSN=0359 1
      SSN=0360 1
      SSN=0361 1
      SSN=0362 1
      SSN=0363 1
      SSN=0364 1
      SSN=0365 1
      SSN=0366 1
      SSN=0367 1
      SSN=0368 1
      SSN=0369 1
      SSN=0370 1
      SSN=0371 1
      SSN=0372 1
      SSN=0373 1
      SSN=0374 1
      SSN=0375 1
      SSN=0376 1
      SSN=0377 1
      SSN=0378 1
      SSN=0379 1
      SSN=0380 1
      SSN=0381 1
      SSN=0382 1
      SSN=0383 1
      SSN=0384 1
      SSN=0385 1
      SSN=0386 1
      SSN=0387 1
      SSN=0388 1

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

Fig. 2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 ( 14 )

FILE=J0001.DDG R2DVP.FORT77<GR2D41>
      C----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      C.TN * CALL GR2D4A(DTB,R(1,3),R(1,2),R(1,3),RNF(2),RNF(3),RNF(4),AC)
      V   IF(CRNF(11,2)*RNF(11,3),NE.0.ODO)
      * CALL GR2D4A(DTB,R(1,1,2),R(1,1,1,3),
      *           RNF(11,2),RNF(11,3),RNF(11,4),THETA(11,4),AC,I1)

      C   DTB=PHI1(2)*PHI12(1)*PHI12(4)*DPHI12(6)
      V   DTB=PHI12*PHI121*PHI124*DPHI126
      C.TN * IF(CRNF(2)*RNF(4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,2),R(1,4),RNF(2),RNF(4),THETA(6),AC)
      * IF(CRNF(11,2)*RNF(11,4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,2),R(1,1,1,4),
      *           RNF(11,2),RNF(11,4),THETA(11,6),AC,I1)

      C   ACC. DUE TO BIAS OF BOND3
      C.TN DTB=PHI13*PHI12*DPHI122*DPHI124*PHI125
      V   DTB=PHI13*DPHI122*PHI124*PHI125
      C.TN * IF(CRNF(1)*RNF(3),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1),R(1,3),RNF(1),RNF(3),RNF(4),THETA(2),AC)
      * IF(CRNF(11,1)*RNF(11,3),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,1),R(1,1,1,3),
      *           RNF(11,1),RNF(11,3),RNF(11,4),THETA(11,2),AC,I1)

      C   DTB=PHI1(3)*PHI12(2)*DPHI12(4)*DPHI12(5)
      V   DTB=PHI13*PHI122*DPHI124*DPHI125
      C.TN * IF(CRNF(2)*RNF(3),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,2),R(1,4),RNF(2),RNF(4),THETA(4),AC)
      * IF(CRNF(11,2)*RNF(11,3),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,2),R(1,1,1,3),
      *           RNF(11,2),RNF(11,3),RNF(11,4),THETA(11,2),AC,I1)

      C   DTB=PHI1(3)*PHI12(2)*DPHI12(4)*DPHI12(5)
      V   DTB=PHI13*PHI122*DPHI124*DPHI125
      C.TN * IF(CRNF(3)*RNF(4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,3),R(1,4),RNF(3),RNF(4),THETA(5),AC)
      * IF(CRNF(11,3)*RNF(11,4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,3),R(1,1,1,4),
      *           RNF(11,3),RNF(11,4),THETA(11,5),AC,I1)

      C   ACC. DUE TO BIAS OF BOND4
      C.TN DTB=PHI1(4)*PHI12(2)*PHI12(5)*DPHI12(6)
      V   DTB=PHI14*DPHI123*DPHI125*DPHI126
      C.TN * IF(CRNF(1)*RNF(4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1),R(1,4),RNF(1),RNF(4),THETA(3),AC)
      * IF(CRNF(11,1)*RNF(11,4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,1),R(1,1,1,4),
      *           RNF(11,1),RNF(11,4),THETA(11,4),AC,I1)

      C   DTB=PHI1(4)*PHI12(3)*DPHI12(5)*DPHI12(6)
      V   DTB=PHI14*DPHI123*DPHI125*DPHI126
      C.TN * IF(CRNF(3)*RNF(4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,3),R(1,4),RNF(3),RNF(4),THETA(5),AC)
      * IF(CRNF(11,3)*RNF(11,4),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,3),R(1,1,1,4),
      *           RNF(11,3),RNF(11,4),THETA(11,5),AC,I1)

      C   ACC. DUE TO BIAS OF BOND5
      C.TN DTB=PHI1(5)*PHI12(2)*PHI12(4)*DPHI12(6)
      V   DTB=PHI15*DPHI123*DPHI125*DPHI126
      C.TN * IF(CRNF(5)*RNF(6),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,5),R(1,6),RNF(5),RNF(6),THETA(6),AC)
      * IF(CRNF(11,5)*RNF(11,6),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,5),R(1,1,1,6),
      *           RNF(11,5),RNF(11,6),THETA(11,6),AC,I1)

      C   ACC. DUE TO BIAS OF BOND6
      C.TN DTB=PHI1(6)*PHI12(2)*PHI12(5)*DPHI12(6)
      V   DTB=PHI16*DPHI123*DPHI125*DPHI126
      C.TN * IF(CRNF(6)*RNF(7),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,6),R(1,7),RNF(6),RNF(7),THETA(7),AC)
      * IF(CRNF(11,6)*RNF(11,7),NE.0.ODO)
      V   CALL GR2D4A(DTB,R(1,1,6),R(1,1,1,7),
      *           RNF(11,6),RNF(11,7),THETA(11,7),AC,I1)

```

Fig.2.10 Modified Subroutine GR2D41 (continued)

```

★ GR2D41 ( 15 )
FILE=J0001.DDGR2DVP.FORT77(GR2D41)
FORTRAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
C C.TN DTB=PHI1(4)*PHI2(3)*PHI1(5)*DPHI12(6) 00036000
C C.TN DTB=PHI14*PHI23*PHI15*D PHI26 00036100
V V C.TN IF(CRNF(2)*RNF(4)-NE-0.0DO) 00036100
C C.TN * CALL GR2D4A(DTB,R(1,2)-R(1,4),RNF(2),RNF(4),THETA(6),AC) 00036200
V V C.TN * CALL GR2D4A(DTB,R(1,2)-R(1,4),RNF(11,4)-NE-0.0DO) 00036300
V V C.TN * CALL GR2D4A(DTB,R(1,2)-R(1,4),RNF(11,2),RNF(11,4),AC,11) 00036200
C C POTENTIAL ENERGY(NON IONIC) 00036400
C C.TN DO 8010 I=1,4 00036500
C C.TN IFCRNFC(1)-NE-0.0DO) RNF(I)=1.0DO 00036600
C C.TN IFCRNFC(1,1)-NE-0.0DO) RNF(C1,1)=1.0DO 00036700
C C.TN IFCRNFC(1,2)-NE-0.0DO) RNF(C1,2)=1.0DO 00036700
C C.TN IFCRNFC(1,3)-NE-0.0DO) RNF(C1,3)=1.0DO 00036700
C C.TN IFCRNFC(1,4)-NE-0.0DO) RNF(C1,4)=1.0DO 00036700
C C.TN ENGPO=ENGPO+( 00036800
C C.TN * RNF(1)* PHI1(1)*PHI2(1)*PHI2(2)*PHI2(3) 00036900
C C.TN *+RNF(2)* PHI1(2)*PHI2(1)*PHI2(4)*PHI2(6) 00037000
C C.TN *+RNF(3)* PHI1(3)*PHI2(2)*PHI2(4)*PHI2(5) 00037100
C C.TN *+RNF(4)* PHI1(4)*PHI2(3)*PHI2(5)*PHI2(6)/2.0D0 00037200
V V ENGPO=ENGPO+( 00036800
V V * RNF(C1,1)* PHI11*PHI121*PHI122*PHI123 00036900
V V *+RNF(C1,2)* PHI12*PHI121*PHI124*PHI126 00037000
V V *+RNF(C1,3)* PHI13*PHI122*PHI124*PHI125 00037100
V V *+RNF(C1,4)* PHI14*PHI123*PHI125*PHI126)/2.0D0 00037200
V V --5500 CONTINUE 00037300
END 00037400
SSN=0435 00037500
SSN=0434 00037600

```

Fig. 2.10 Modified Subroutine GR2D41 (continued)

```
TWC(4 2 8 1 4) SRP
//FORTEXVP EXEC FOLRTEXVP, SO=' J0001.DDGR2DVP',
//      A=' ELM(*),NOS'. OPT=F
//GRL2      EXEC LKEDCTEX, LM=' J0001.DDGR2DVP', UNIT=' D0340'
//
```

Fig.2.11 JCL for Making Load-module

```
TWC(4 2 8 1 4) SRP CLASS(0)
//CR2GO EXEC LMGO=' J0001.DDGR2DVP'
//FT05F001 DD DSN=J0001.DIA.DATA(PART2), DISP=SHR
//FT18F001 DD DSN=J0001.DIA18T.DATA, DISP=SHR
//FT19F001 DD DSN=J0001, DIA19T.DATA, DISP=(NEW, CATLG),
//           SPACE=(TRK,(100,100),RLSE), UNIT=TSSWK
```

Fig.2.12 JCL for Execution

```
SUBROUTINE GR2D41(N)
IMPLICIT REAL*8 ( A-H, O-Z)
C
C. TN  VECTORIZATION FOR PARAMETER
PARAMETER(LV=30)                                     ⇔
COMMON /COMBIN/KK, III1(50), III4(50)
```

Fig.2.13 Declaration of Work Area

3. STREAM V3.1 のベクトル化

3次元非定常圧縮性流体用解析コード STREAM V3.1 は、非圧縮性流体のみを取り扱うことができた前バージョン Ver.2.6 を改良して、圧縮性の流体も取り扱うことができるようとしたコードである。

前バージョンは、すでに情報システムセンターにおいてベクトル化が行われており [1] 高速な計算が可能となっていたが、バージョンアップにともなって新たなサブルーチンなどが付け加えられ、また前バージョンでベクトル化された部分も元に戻ったため、再度ベクトル化を行う必要性が生じた。

本章では、富士通製スーパーコンピュータ VP-2600 とコンバイラ FORTRAN77/EX V12 [9] を用いた高速計算のために、STREAM V3.1 を母体として日本原子力研究所において改良作業が行われつつあるコード [4-8] を対象として行ったベクトル化について述べる。

3.1 基礎方程式

STREAM V3.1 が扱う方程式は、非圧縮性の流体を扱う場合と圧縮性の流体を扱う場合で異なる。本ベクトル化作業では圧縮性流体を扱う部分のみを作業の対象としたため、ここでは圧縮性流体を扱う部分について述べる。

(1) 質量の保存式

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i} \rho u_i = 0 \quad (3.1)$$

(2) 運動量の保存式 ($i = 1 \sim 3$)

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial u_j \rho u_i}{\partial x_j} = \frac{\partial \sigma_{ij}}{\partial x_j} + \rho g_i \quad (3.2)$$

(3) エネルギーの保存式

$$\frac{\partial \rho H}{\partial t} + \frac{\partial u_i \rho H}{\partial x_j} = -\nabla \cdot q^{(d)} + \frac{\partial p}{\partial t} + \frac{\partial u_i p}{\partial x_j} + \sigma_{ij} \frac{\partial u_i}{\partial x_j} + \frac{\partial}{\partial x_j} K \frac{\partial T}{\partial x_j} + \dot{q} \quad (3.3)$$

(4) 乱流エネルギー・乱流消失率の式 ($k - \epsilon$ 方程式)

$$\frac{\partial \rho k}{\partial t} + \frac{\partial u_i \rho k}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\frac{\mu_t}{\sigma_k} \frac{\partial k}{\partial x_i} \right) + \rho (G_s - G_{s1} - G_{s2} - G_{s3} - \epsilon) \quad (3.4)$$

$$\frac{\partial \rho \epsilon}{\partial t} + \frac{\partial u_i \rho \epsilon}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\frac{\mu_t}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial x_i} \right) + C_1 \frac{\epsilon}{k} (G_s - G_{s1} - G_{s2} - G_{s3} - \epsilon) - C_2 \frac{\rho \epsilon^2}{k} \quad (3.5)$$

(5) 拡散物質の保存式

$$\frac{\partial \rho C}{\partial t} + \frac{\partial u_j \rho C}{\partial x_j} = \frac{\partial}{\partial x_j} \rho D_m \frac{\partial C}{\partial x_j} + \rho d \quad (3.6)$$

(6) 状態方程式

$$p = \rho RT \quad (3.7)$$

これらの式に現われる変数の意味は次の通りである。

x_i	:	位置座標	(m)
u_i	:	x_i 方向の流体速度	(m/s)
t	:	時間	(s)
ρ	:	流体あるいは固体の密度	(kg/m ³)
p	:	流体の圧力	(N/m ²)
μ	:	粘性係数	(kg/m · s)
		($= \mu_l + \mu_t$: 分子粘性係数 + 涡粘性係数)	
σ_{ij}	:	応力テンソル ($= \mu(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) - (p + \frac{2}{3}\mu\frac{\partial u_k}{\partial x_k}\delta_{ij})$)	
H	:	比エンタルピー	(J/kg)
g_i	:	加速度	(m/s ²)
T	:	流体あるいは固体の温度	(°C)
K	:	熱伝導率	(J/m · s · °C)
\dot{q}	:	発熱量	(J/m ³ · s)
k	:	乱流エネルギー	(m ² /s ²)
ϵ	:	乱流消失率	(m ² /m ³)
C	:	拡散物質の濃度	(1)
D_m	:	拡散係数	(m ² /s)
\dot{d}	:	拡散物質の発生項	(1/s)
R	:	ガス定数	(J/kg · °K)
$\nabla \cdot q^{(d)}$:	濃度差によるエンタルピー輸送項	

また、本コードは多成分気体としてヘリウム、酸素、一酸化炭素、二酸化炭素、窒素、空気、及びこれらの混合ガスを対象としており、これらの物性値(定圧比熱、熱伝導率、粘性係数、有効多成分拡散係数)を計算するサブルーチンが追加されている。これら物性値の推算式を以下に示す。

(1) 定圧比熱 C_p [J/kg · K]

- i 成分気体

$$C_{pi} = (A_{ci} + B_{ci}T + C_{ci}T^2 + D_{ci}T^3) * 10^3 \quad (3.8)$$

- 混合気体

$$C_{pm} = \sum_{i=1}^n (C_{pi}X_i) \quad (3.9)$$

(2) 热伝導率 λ [W/m·K]

- i 成分気体

$$\lambda_i = (1.32C_{vi}^o + 3.52)\mu_i/M_i \times 4.186 \times 10^3, \quad (3.10)$$

$$C_{vi}^o = C_{vi}M_i/4.186,$$

$$C_{vi} = A_{c2} + B_{c2}T + C_{c2}T^2 + D_{c2}T^3$$

- 混合気体

$$\lambda_m = \sum_{i=1}^n (X_i \lambda_i / \sum_{j=1}^n X_j A_{ij}), \quad (3.11)$$

$$A_{ij} = \frac{\{1 + (\lambda_i/\lambda_j)^{1/2}(M_j/M_i)^{1/4}\}^2}{\{8(1 + M_i/M_j)\}^{1/2}}$$

(3) 粘性係数 μ [Pa·S]

- i 成分気体

$$\mu_i = 26.69(M_i T)^{1/2}/\sigma^2 \Omega_v \times 10^{-7}, \quad (3.12)$$

$$\Omega_v = \frac{A_v}{T^{*B_v}} + \frac{C_v}{\exp(D_v T^*)} + \frac{E_v}{\exp(F_v T^*)},$$

$$T^* = kT/\epsilon_i$$

- 混合気体

$$\mu_m = \sum_{i=1}^n (X_i \mu_i / \sum_{j=1}^n X_j \phi_{ij}), \quad (3.13)$$

$$\phi_{ij} = \frac{\{1 + (\mu_i/\mu_j)^{1/2}(M_j/M_i)^{1/4}\}^2}{\{8(1 + M_i/M_j)\}^{1/2}}$$

(4) 有効多成分拡散係数 D_{ij} [M²/s]

- 2 成分拡散係数

$$D_{ij} = 1.858 \times 10^{-7} T^{3/2} \frac{\{(M_i + M_j)/(M_i M_j)\}^{1/2}}{P \sigma_{ij}^2 \Omega_d}, \quad (3.14)$$

$$\sigma_{ij} = (\sigma_i + \sigma_j)/2,$$

$$\Omega_d = \frac{A_d}{T^{*B_d}} + \frac{C_d}{\exp(D_d T^*)} + \frac{E_d}{\exp(F_d T^*)} + \frac{G_d}{\exp(H_d T^*)},$$

$$T^* = kT/\epsilon_{ij},$$

$$\epsilon_{ij} = (\epsilon_i \epsilon_j)^{1/2}$$

- 混合気体

$$D_{im} = \left\{ \sum_{j=1, j \neq i}^n (X_j/D_{ij}) \right\}^{-1} \quad (3.15)$$

ここで、下つき添字の i は成分気体の種類を、 m は混合平均値を示す。また、式中に現われる変数の意味は次の通りである。

$A \sim D$: 定数
X	: 質量分率
M	: 分子量
C_v	: 定容比熱
Ω_d	: 衝突積分値
σ_{ij}	: 原子間距離
k	: ボルツマン定数
P	: 圧力
ϵ	: 特性エネルギー
T^*	: 特性温度

3.2 ベクトル化方法

本コードには実行時に各種のオーバーヘッドを非常に多く発生する部分が含まれている。このため、次のような手順でベクトル化作業を行った。

- i) まず、各種のオーバーヘッドを削減する。
- ii) オーバーヘッドの削減後のソースプログラムを用いて ANALYZER [11] による動的挙動解析を行い、計算コストの大きいルーチンを選び出す。
- iii) ii) で選び出した各ルーチンについて、ベクトル化が可能であるかを調べ、可能なものについてベクトル化作業を行う。

以下、具体的な作業内容について述べる。

3.2.1 各種オーバーヘッドの削減

(3.8) 式から (3.15) 式に示した各物性値を計算するルーチンの構成の概要を Fig. 3.1 に示す。この図において、上位ルーチン中の 3 重 DO ループは 3 次元解析空間上のメッシュ点を示し、この中から呼び出されるファンクションで、指定されたメッシュ点における物性値を計算する。また、混合気体の物性値を計算する際に総和計算を行う必要があるため、この下位のファンクション中には、気体の種類数を回転数とする DO ループが存在している。

この図に示したルーチンを実行した場合、本作業で用いた入力データでは 3 次元解析空間の一辺がそれぞれ数 10 程度であるため、下位のファンクションは数千から数万回程度呼び出されることになる。また、このルーチンは各時間ステップごとに計算される。したがって、このファンクション呼び出し時のオーバーヘッドはかなり大きなものとなる。

このルーチンをベクトルコンパイルした場合、ベクトル化の対象となるのは下位のファンクション中の DO ループである。しかし、この DO ループの回転数は扱う気体の種類数であり、ベクトル長は非常に短い。このような場合にはベクトル実行による高速化が得られず、むしろベ

クトルバイオライン立ち上げに要するオーバーヘッドのために遅くなる。このオーバーヘッドがファンクションを呼び出す度に発生するため、非常に大きなものとなる。

以上の点を改善するため、Fig. 3.1に示したプログラム構造に手を加えて、Fig. 3.2に示す構造に変更した。すなわち、上位ルーチンの3重DOループ内の処理を一つのサブルーチンにまとめるこことによって、外部手続きの呼び出し回数を大幅に削減している。また、3重DOループを下位ファンクション中のDOループの内側に配置している。このとき、3重DOループの最内ループが自動ベクトル化の対象となり、ベクトル長が数10程度と長くなるため、ベクトル処理効率が向上する。

3.2.2 多重DOループの一重化

STREAMコードは、流体の流速や温度などの方程式を解く際に、3次元空間の7点差分による7重対角行列を作成する。この行列を作成する部分は、非常に大きな3重DOループから構成されている。この3重DOループの構成をFig. 3.3に示す。この図に示した3重DOループは、その最内ループがコンパイラによる自動ベクトル化の対象となるが、この時のベクトル長は解析空間の1方向のメッシュ数に等しく、数十程度と比較的小さいためベクトル実行による効果が小さい。

このため、Fig. 3.4に示すように手を加えて3重DOループを一重化し、一重化後のDOループが自動ベクトル化の対象となるようにした。このときのベクトル長は、解析空間全体のメッシュ数にほぼ等しく、非常に大きなベクトル化効果が期待できる。

3.2.3 外部手続きのインライン展開

DOループの内側から、サブルーチンや外部関数が呼び出されている時には、このDOループは完全なベクトル化が行われない。このような場合のベクトル化の例として、3.2.1ではDOループ内の処理をひとつのサブルーチンにまとめるこことによってベクトル化を行った。この他に、インライン展開を利用する方法がある。下位の手続きの内容をその呼び出し箇所に書き下した後、特にベクトル化を阻害する要因がある場合にはこれを取り除くことによって、そのDOループが自動ベクトル化されるようになる。このベクトル化作業では、下位の手続きの内容を呼び出し箇所に書き下さず、次のようなコンパイルオプションを指定してコンパイラのインライン展開機能[10]を利用した。

```
INLINE(INTERNAL(120,AAA,BBB,...))
```

この場合にも、インライン展開後のDOループの内容にベクトル化を阻害する要因が存在しなければ、自動ベクトル化が行われる。

3.2.4 Hyper Plane法

STREAMコードは、JACOBI法、SOR法、MICCG法、及びILUCR法による4種類の7重対角行列のソルバを持っており、解くべき方程式に応じていずれかを選択できるようになっている。この4種類のソルバの内、SOR法、MICCG法、ILUCR法によるソルバの反復計算

部分は、回帰参照のため自動ベクトル化が行われていなかった。これらの部分に Hyper Plane 法を適用して回帰参照を解消し、自動ベクトル化を促進した。

3.3 ベクトル化の結果

ベクトル化前のソースプログラムを用いて実行した場合及びベクトル化を施した後のソースプログラムを用いて実行した場合の実行時間を、それぞれ Table 3.1, 3.2 に示す。これらの表から、ベクトル化前において最も実行時間の速いスカラ実行（コンパイル時の最適化レベル E）と比較して、6.8～14.8 倍程度の高速化が得られている。ただし、ここでは述べなかったが、ベクトル化作業中に計算式に若干の変更を加えてあるため、この速度比は厳密なものではない。

なお、この時用いた入力データの概要は次の通りである。

TA00, TA01

- 直径 0.1m, 高さ 0.2m の円筒（メッシュ数 $15 \times 10 \times 20$ の円柱座標系）。
- 円筒内にはヘリウム、円筒外は空気で満たされている。
- 圧力分布をつけるために初期計算を行う（入力データ名 TA00, 反復回数 10）。
- 初期計算の後、円筒の上部に直径 0.015m の穴を空け、その後の過渡状態を計算する（入力データ名 TA01, 反復回数 300）。

TB00, TB01

- 内直径 0.1m, 外直径 0.3m, 高さ 0.2m のドーナツ状円筒（メッシュ数 $25 \times 30 \times 30$ の円筒座標系）。
- 円筒内はヘリウム、円筒外は空気で満たされている。
- 圧力分布をつけるために初期計算を行う（入力データ名 TB00, 反復回数 5）。
- 初期計算の後、円筒の上部に穴を空け、その後の過渡状態を計算する（入力データ名 TB01, 反復回数 100）。

TD00, TD01

解析対象などは TA00, TA01 と全く同様である。ただし、TA00, TA01 では ILUCR 法による行列ソルバを用いているのに対して、TD00, TD01 では SOR 法によるソルバを用いる。

Table 3.1 CPU time of the original version in seconds.

(a) スカラ実行(コンパイル時の最適化レベルE)

入力データ	TA00	TA01	TB00	TB01	TD00	TD01
CPU	8.41	264.21	0.01	540.69	8.03	283.01

(b) ベクトル実行(コンパイル時の最適化レベルE)

入力データ	TA00	TA01	TB00	TB01	TD00	TD01
CPU	9.76	295.82	34.93	611.45	9.51	317.50
VU	4.78	130.38	17.52	294.41	4.74	129.21

Table 3.2 CPU time of the vectorized version in seconds.

(a) スカラ実行(コンパイル時の最適化レベルE)

入力データ	TA00	TA01	TB00	TB01	TD00	TD01
CPU	7.36	248.03	29.15	553.96	10.27	365.46

(b) ベクトル実行(コンパイル時の最適化レベルE)

入力データ	TA00	TA01	TB00	TB01	TD00	TD01
CPU	1.04	23.04	2.43	36.45	1.18	29.52
VU	0.47	15.61	1.64	28.41	0.63	22.21

```

SUBROUTINE XXXXXX( ... )
<中略>
DO 150 K=1,KM
DO 150 J=1,JM
DO 150 I=1,IM
<中略>
EXCO(I,J,K,1)=YYYYYY( ... ,I,J,K) ! ==> 外部手続き呼び出しの
! オーバーヘッド
<中略>
150 CONTINUE
<中略>
RETURN
END
C
FUNCTION YYYYYY( ... ,I,J,K)
<中略>
DO 1000 M=1,ITEMC ! ==> ベクトルバイブライン立ち上げのオーバーヘッド
! (ベクトル長は最大で 6 )
<中略>
1000 CONTINUE
<中略>
RETURN
END

```

Fig.3.1 Original program structure for calculations of physical properties.

```

SUBROUTINE XXXXXX( ... )
<中略>
CALL YYYYYY( ... ,I,J,K,EXCO)
<中略>
RETURN
END
C
SUBROUTINE YYYYYY( ... ,I,J,K,EXCO)
DIMENSION EXCO( ... )
<中略>
DO 1000 M=1,ITEMC
    DO 150 K=1,KM
    DO 150 J=1,JM
    DO 150 I=1,IM
        <中略>
        EXCO(I,J,K,1)= ...
150    CONTINUE
1000 CONTINUE
<中略>
RETURN
END

```

Fig.3.2 Improved program structure for calculations of physical properties.

```

DO 100 K=1,KM
<中略>
DO 100 J=1,JM
<中略>
DO 100 I=2,IM
<中略>
100 CONTINUE

```

Fig.3.3 Example of a threefold DO loop.

```

C      DO 100 K=1,KM
      IM1=IM-1
      IJM=IM1*JM
*VOCL LOOP,NOVREC
      DO 100 IJK=1,IM1*JM*KM
          K = (IJK-1)/IJM+1
          IJ= IJK-(K-1)*IJM
          J = (IJ -1)/IM1+1
          I = IJ -(J-1)*IM1+1
          <中略>
C      DO 100 J=1,JM
          <中略>
C      DO 100 I=2,IM
          <中略>
100 CONTINUE

```

Fig.3.4 Vectorization of the threefold DO loop in Fig.3.3 by converging the loop into a single DO loop.

4. CELLA のベクトル化

CELLA はセル・オートマトンモデルを用いた二次元流体シミュレーションコードである。セル・オートマトンとは、六角格子上のブランチと格子点に 1 ビット（仮想粒子：有り or 無し）の仮想粒子があるモデルである。時間（離散）ステップ毎に粒子は 1 格子距離だけ進み、1 つの格子点に集まつた粒子はあらかじめ決められた規則（有限の反応テーブル）に基づき散乱される。粒子間の衝突は、全ての衝突パターンが記憶されている反応テーブルを参照することにより行なわれる。

ここでは CELLA に対して行ったベクトル化作業について述べる。

4.1 動的解析

実行時における、計算コストの分布を調べるために、ANALYZER [11] 及び、TOP10EX [12] を使用し、各サブルーチンについて動的解析を行った。

動的解析のための入力データは、上流側境界から粒子投入、下流側境界は真空、サンプリング回数を 10 回、サンプリング間のシミュレーション回数を 10 回とし、リスタートデータを出力しない初期状態からの実行である CASE1 のデータを用いた。計算コスト分布を Table 4.1 に示す。

この表より、上位 3 個のサブルーチン (MOVE, SAMPLE, COLLIS) の計算コストの合計が 99.7% となるので、これらをベクトル化の対象ルーチンとした。

また、CELLA コードの作画部分を除いた木構造を Fig.4.1 に示す。

4.1.1 サブルーチン MOVE

まず、全体の 75.9% のコストがかかっているサブルーチン MOVE について調べる。

サブルーチン MOVE は、粒子の移動処理を行なっており、格子点間の粒子の移動は流れ場の部位ごとに設定されたテーブルを参照して行なわれている。参照するテーブルは、部位ごとに粒子の存在するブランチを示すテーブル* と任意の格子点の周囲の格子点番号を算出するテーブル† の 2 つを行い、移動後の粒子の置き換えは、流れ場中の全ての格子点についての移動処理が終了してから行なわれる。Fig.4.2, Fig.4.3 に概略流れ図を示す。

サブルーチン MOVE の詳細解析の結果、流れ場の計算を行なっている 3 重 DO ループに最もコストがかかっていることがわかった。また、上流側境界に存在する粒子の検出ならびに移動処理を行なっている 2 重 DO ループは、ベクトル長が短く効率が悪い。よって、この 2 つの多重 DO ループをベクトル化対象とした。Fig.4.6 にその部分の詳細解析結果を示す。

4.1.2 サブルーチン COLLIS

次に、全体の 21.6% の計算コストがかかっているサブルーチン COLLIS について調べる。

* Fig.4.2,4.3 概略流れ図における※ 1 ステップ

† Fig.4.2,4.3 概略流れ図における※ 2 ステップ

このルーチンでは粒子間の衝突処理を行なっている。粒子間の衝突は、反応テーブルで示されている格子点の粒子の状態と一致した場合に起きる。反応テーブルは、全ての粒子の状態に対応した反応が格納されており、このテーブルを参照するだけで衝突後の粒子の状態が決定される。しかし、反応によっては2通りの衝突後の粒子の状態からランダムに決定する場合もある。また、反応テーブル1,2は格子点の値をアドレスとして参照される。Fig.4.4に概略流れ図を示す。

	衝突無	衝突有
反応テーブル1	格子点の値のまま	反応後の値
反応テーブル2	0	反応後の値

サブルーチン COLLIS の詳細解析の結果、ベクトル長 2046, DO 変数 IW で回る DO 100 ループが最も計算コストが高く、ベクトル化されていないため、このループをベクトル化対象ループとした。Fig.4.10にその部分の詳細解析結果を示す。

4.1.3 サブルーチン SAMPLE

サブルーチン SAMPLE は、全体の 2.2% の計算コストがかかっている。このルーチンでは、任意のシミュレーション・サイクル（サンプリング間隔）毎のサンプリング・セル内の格子点情報の獲得を行なっている。

サンプリングは、シミュレーション・サイクルにおける衝突と移動の処理の後で行なわれるため粒子が流れ場の外へ存在する可能性がある。そのため、サンプリングの直前で境界処理を行ない、それらの粒子を流れ場へ引き戻してからサンプリングを行なっている。Fig.4.5に概略流れ図を示す。

サブルーチン SAMPLE の詳細解析の結果、DO 変数 M で回るブランチのループ DO 100, DO 200, DO 600 はベクトル化されているが、ベクトル長が 7 と短く、ベクトル処理に要する時間に対し、ベクトル処理のためのオーバーヘッドに要する時間が無視できなくなり、このままでは実行効率が悪い。よってこの部分をベクトル化対象ループとした。Fig.4.12にその部分の詳細解析結果を示す。

4.2 ベクトル化方法

4.2.1 ループの入れ換え

サブルーチン MOVE では、上側壁境界の処理と下側壁境界の処理及び、流れ場の粒子の移動処理を行なっているそれぞれの2重 DO ループにおいて、最内ループはテーブル参照ループであり、ベクトル長は最大でも 6 と短い。特にコストが高い流れ場の3重 DO ループは、ベクトル長 2047 の Y 軸方向の格子点列及び、ベクトル長 2046 で X 軸方向の格子点列のループ DO 550、そして、最内ループはベクトル長 6 の参照テーブルの参照ループ DO 500 から構成されている (Fig.4.6)。

ここで、反応テーブルの参照に用いられている DO 500 ループを最外に出してもループ内の粒子の検出及び、移動後の粒子の格納ステップには影響しない。そこで、このループを 3 重ループの最外に移動した。その結果、ベクトル化対象の DO ループは、ベクトル長が 2046、DO 変数 IW で回るループとなる。

ただし、このままではコンバイラは、配列 GWK に関する定義参照関係が不明のため、配列を回帰データとみなしてベクトル化しない。しかし、実際はその配列に関する定義引用関係がベクトル化可能であるとわかっているので、NOVREC 情報を含む最適化制御行 [10] を原始プログラムに挿入することにより、ベクトル化の促進を図った。

同様な処理により、サブルーチン SAMPLE もベクトル化処理効率の向上を図った。DO ループの入れ換えにより、ベクトル化した例を Fig.4.7 に示す。

4.2.2 DO 文の書き下し

サブルーチン MOVE では、上流側境界に存在する粒子の検出、ならびに移動処理を行っている 2 重ループ DO 150, DO 100 において、最内ループ DO 100 のベクトル長は、その上の IF 文を含む処理によって決まる DO 変数 JS, JE によって決定される。しかし、ベクトル長は最大でも 5 であり、ベクトル化効率が悪い (Fig.4.8)。

そこで、DO 100 ループを書き下し、変数 I を定義する代入文を DO 100 の直後へ移動した。しかし、このままではコンバイラは、流れ場の格子点情報を格納している作業用配列 GWK の定義引用関係が不明のため、回帰データの可能性があると判断し、ベクトル化しない。そこで、配列 GWK は回帰データでないことがわかっているので、NOVREC 情報を含む最適化制御行を原始プログラムに挿入することにより、ベクトル長 2048 の DO 150 でベクトル化した。また、下流側境界の処理を行なっている部分及び、サブルーチン VACUUM でも同様な処理を施した。

DO ループの書き下しによりサブルーチン MOVE をベクトル化した例を Fig. 4.9 に示す。

4.2.3 DO ループの分割

サブルーチン COLLIS の DO 100 ループ内では、乱数発生ルーチン INRGEN を CALL している (Fig.4.10)。CALL 文などの外部手続き文はベクトル化対象外であり、このままではベクトル化できない (Fig. 4.4, 4.10 参照)。このように、DO ループ内に CALL 文が存在する場合、CALL 文の前後でループを分割するか、下位のサブルーチンに DO ループを引き込む、または呼び出されるルーチンをインライン展開するなどプログラムを変更する必要がある。

今回の作業では、リストを作成し、CALL 文の前後で DO ループを分割する前者の方法をとった。

まず、Fig.4.10 の SSN 383 の反応テーブル 2 の粒子の衝突の有無を判定する IF 文、

```
IF( REFTL2(GRID(I)) . GT . 0 ) THEN
```

において、この IF 文が真の場合の時の流れ場中の粒子番号の値 I を格納しておくために、作業用配列 IFLAG を作成しておく。この作業用配列の大きさは、サンプリングセル数以上なければならないため 1024 とした。そして、この作業用配列の添字として DO 変数 IW を使用し、流れ

場の粒子番号の値を格納し、フラグの変わりにする。同時に、この IF 文を通った回数を作業用変数 NRANSU でインクリメントしていく。この変数は後に乱数の個数が足りているかどうかの判定に使われる。IF 文が偽の場合には、作業用配列 IFLAG の値は 0 とし、NRANSU はカウントしない。以上で 1 段めのループを閉じる。

次に、乱数が足りているかどうかを判定し、足りなければ INRGEN() を呼び出す部分 (Fig.4.10 SSN 385 の IF 文),

```
IF ( L.GT.IRMAX) THEN
    CALL INRGEN(2,IRMAX)
    L = L + 1
ENDIF
```

を独立させ、

```
IF ( L + NRANSU .GT. IRMAX ) THEN
    CALL INRGEN(2,IRMAX)
    L = L + 1
ENDIF
```

のように変更した。乱数の数は先ほどカウントした NRANSU を使用する。そして、反応テーブル 2 を使用し粒子の衝突の処理を行うループを新たに作成する。ここで必要となる粒子番号は、先ほど作成したリスト配列 IFLAG に格納されているものを使い、L をカウントする。

このようにプログラムを変更することにより、X 方向の格子点列のループをベクトル化できる。ベクトル化した例を Fig.4.11 に示す。

4.3 計算結果の評価

CELLA コードは、計算結果を図形出力するため、ベクトル化による計算結果の評価は、ベクトル化対象の各サブルーチンで、定義される変数および配列を書き出し、オリジナル版の結果と照合することにした。照合の結果、それらの変数及び配列は、ベクトル化対象の全てのサブルーチンにおいて全桁一致した。

4.4 ベクトル化効果

ベクトル化の効果を測定するために、VP2600 上でオリジナル版及びベクトルチューニング版それぞれのスカラモードとベクトルモードの CPU, REGION, I/O を測定した。実行には CASE1 のデータと、初期条件及び境界条件は同じく、サンプリング回数とサンプリング間のシミュレーション回数をそれぞれ 20 回にした CASE2 のデータを用いた。

Table.4.2 には各モードにおける実行時間、領域、I/O 回数、速度向上比、そしてベクトル化率を、Table.4.3 にベクトルチューニング後のコスト分布をそれぞれ示す。速度向上率及びベクトル化率はそれぞれ次の式で表される。

$$\begin{aligned}\text{ベクトル化率}(V) &= ts2 / (ts1 + ts2) \\ &= (SCPU - (VCPU - VU)) / SCPU\end{aligned}$$

$$\begin{aligned}\text{ベクトル/スカラ速度比} &= ts2 / VU \\ &= (SCPU - (VCPU - VU)) / VU\end{aligned}$$

SCPU : プログラムをスカラーモードで実行した時の CPU 時間

VCPU : プログラムをベクトルモードで実行した時の CPU 時間

VU : ベクトル処理可能な部分のベクトル CPU 時間

ts1 : ベクトル処理できない部分の CPU 時間

ts2 : ベクトル処理可能な部分のスカラ CPU 時間

4.2より、CASE1 データにおいて、オリジナル版スカラ計算に対し、チューニング版ベクトル計算に要した時間は約 16 分の 1 であり、速度比は 15.9 倍になる。またベクトル化率は、オリジナル版ベクトル計算の 29.51% に対し、チューニング版ベクトル計算は 99.72% になった。これに対し、CASE2 での速度比は 19.07 倍であり、ベクトル化率は 99.89% となった。

4.5 まとめ

今回のベクトル化作業において、CELLA コードの速度比は 15 ~ 16 倍になった。しかし、今後さらなる高速化が望まれた場合、改良すべき点はまだある。その一つに多重ループの一重化があげられる。サブルーチン MOVE, SAMPLE では、最内ループであるテーブル参照ループと、空間のループの入れ換えのみ行ったが、それでもかなりの効果が上がっているため、それ以上の変更は行なわなかった。しかし、これらのループを一重化できれば、さらなる高速化が望めるであろうが、アルゴリズムを大幅に変更する必要がある。また、サブルーチン MOVE で行った DO ループの展開も、IF 文のネストが深くなり、IF 文の処理のためのオーバーヘッドがかかってくる。しかし、FORTRAN77 EX/VP コンバイラと VP2600 は、IF 文を含む DO ループのベクトル処理性能が優れているため、現段階においてもかなりの高速化がなされている。今後、アルゴリズムを変更し、IF 文を少なくすることができれば、さらなる高速化が実現可能と考える。

RANK	NAME	V-LENG	V-RATE	V-EFFECT	V-COST	% V-LENG	% V-RATE	% V-EFFECT	% V-COST
1	MOVE	8	16.2	1.1-	1.1	.4339E11	75.9	#####	75.9
2	COLLIS	3069	56.0	2.2-	2.2	.1237E11	21.6	#####	21.6
3	SAMPLE	9	94.2	2.0-	5.0	.1285E10	2.2	#	#
4	VACUUM	2	0.0	1.0-	1.0	29994595	0.0		
5	WALL	2	0.0	1.0-	1.0	23893678	0.0		
6	STARTP	29	4.7	1.0-	1.0	429235	0.0		
7	INRGEN	2008	99.9	58.5-	78.4	195986	0.0		
8	VRAMF	2008	99.9	58.7-	78.7	195386	0.0		
9	GRAPH1	229	31.1	1.4-	1.4	143404	0.0		
10	ALTSSET	41	96.6	9.0-	12.3	42129	0.0		
11	ADDPAR	683	99.8	56.6-	74.9	39383	0.0		
12	GRAPH2	30	93.5	6.9-	8.8	21613	0.0		
13	INISET	7	96.6	2.2-	5.8	15128	0.0		
14	PROFF	512	99.9	58.3-	78.0	1718	0.0		
15	SIMULA	1	0.0	1.0-	1.0	1400	0.0		
16	BOUND	1	0.0	1.0-	1.0	1210	0.0		
17	VRAMFI	5000	99.9	60.0-	81.1	1148	0.0		
18	MAIN	9	0.0	1.0-	1.0	1134	0.0		
19	GRAPH3	1	49.3	0.6-	0.9	820	0.0		
20	MYCONT	6	54.9	1.8-	2.0	263	0.0		
21	PRMSET	1	0.0	1.0-	1.0	69	0.0		
22	CDREAD	1	0.0	1.0-	1.0	27	0.0		
23	BITWRT	0	0	0	0	0	0		
24	BLOCKD	0	0	0	0	0	0		
25	BOUWRT	0	0	0	0	0	0		
26	CYCLIC	0	0	0	0	0	0		
27	GWKWR	0	0	0	0	0	0		
28	MSINT	0	0	0	0	0	0		
29	MSMAN	0	0	0	0	0	0		
30	MSUNI	0	0	0	0	0	0		
31	RFLIN	0	0	0	0	0	0		
32	RFLOUT	0	0	0	0	0	0		
33	SEARCH	0	0	0	0	0	0		
	(TOTAL)		33.7	1.4-	1.4	.5711E11	100.0		

Table 4.1 Dynamic behavior of the vectorized of original CELLA code.

Table 4.2 Vectorization Effect of CELLA.

		CASE1	CASE2	Compiler Options etc.
CPU time CPU time	Original Scalar	8m35.89s	33m08.93s	FORTRAN77EX Ver.12
	Original Vector (VU time)	6m40.62s (0m36.95s)	25m35.35s (1m34.17s)	OPTIMIZE(E)
	Tuning Scalar	8m40.24s	33m37.54s	
	Tuning Vector (VU time)	0m32.45s (0m31.02s)	1m59.99s (1m56.87s)	FORTRAN77EX/VP Ver.12
Region size	Original Scalar	34160 Kbyte	34160 Kbyte	OPTIMIZE(E)
	Original Vector	34600 Kbyte	34600 Kbyte	
	Tuning Vector	34204 Kbyte	34204 Kbyte	
I/O count	Original Scalar	1354	1377	
	Original Vector	2429	2449	
	Tuning Vector	2413	2421	
V-Effect	* OrigS/OrigV	1.29	1.49	
	* OrigS/TuneV	15.90	19.07	
	* OrigV/TuneV	12.35	12.79	
V-Rate	Original Version	29.51%	27.53%	
	Tuning Version	99.72%	99.86%	

* OrigS:Original Scalar, OrigV:OriginalVector, TuneS:Tuning Scalar, TuneV:Tuning Vector

Table 4.3 Dynamic behavior of the vectorized of tuning CELLA code.

RANK	NAME	V-LENG	V-RATE	V-EFFECT	V-COST	%
1	MOVE	2727	99.9	59.5-	80.3	.1325E10
2	COLLIS	2728	99.9	58.9-	79.2	.4422E09
3	SAMPLE	72	98.5	18.6-	25.3	.2098E09
4	STARTP	29	4.7	1.0-	1.0	.4292E35
5	VACUUM	1366	99.9	60.2-	81.4	.3259E15
6	WALL	1365	99.9	60.1-	81.3	.2649E34
7	INRGEM	2008	99.9	58.5-	78.4	.1959E6
8	VRAMF	2008	99.9	58.7-	78.7	.1953E6
9	GRAPH1	229	31.1	1.4-	1.4	.1434E4
10	ALTSET	41	96.6	9.0-	12.3	.4212E9
11	ADDPAR	683	99.8	56.6-	74.9	.3938E3
12	GRAPH2	30	93.5	6.9-	8.8	.2161E3
13	INISET	7	96.6	2.2-	5.8	.1512E8
14	PROPF	512	99.9	58.3-	78.0	.1718E0
15	SIMULA	1	0.0	1.0-	1.0	.1400E0
16	BOUND	1	0.0	1.0-	1.0	.1210E0
17	VRAMFI	5000	99.9	60.0-	81.1	.1148E0
18	MAIN	9	0.0	1.0-	1.0	.1134E0
19	GRAPH3	1	49.3	0.6-	0.9	.820E0
20	MYCONT	6	54.9	1.8-	2.0	.263E0
21	PRMSET	1	0.0	1.0-	1.0	.69E0
22	CDREAD	1	0.0	1.0-	1.0	.27E0
23	BITWRT	0	0	0	0	0
24	BLOCKD	0	0	0	0	0
25	BOUWRT	0	0	0	0	0
26	CYCLIC	0	0	0	0	0
27	GWKWR	0	0	0	0	0
28	MSINT	0	0	0	0	0
29	MSMAN	0	0	0	0	0
30	MSUNI	0	0	0	0	0
31	RFLIN	0	0	0	0	0
32	RFLOUT	0	0	0	0	0
33	SEARCH	0	0	0	0	0
	(TOTAL)	99.9	55.0-	74.2	.1978E10	100.0

```

0001  MAIN-----+CDREAD
      !
0002      +-+PRMSET
      !
0003      +-+INISET
      !
0004      +-+VRAMFI
      !
0005      +-+RFLIN
      !
0006      +-+SIMULA----+ADDPAR-----INRGEN----+VRAMF
      !      !
      !
0007      !      !      +-+?RANU2
      !      !
      !
0008      !      !      +-+MSMAN----+MSINT
      !      !
      !
0009      !      !      +-+MSUNI
      !
      !
0010      !      +-+BOUND----+VACUUM
      !      !
      !
0011      !      !      +-+CYCLIC
      !      !
      !
0012      !      !      +-+WALL
      !
      !
0013      !      +-+COLLIS-----INRGEN---> 0006
      !
      !
0014      !      +-+MOVE
      !
      !
0015      +-+SAMPLE-----BOUND----> 0010
      !
      !
0016      +-+RFLOUT
      !
      !
0017      +-+PROPFF
      !
      !

```

Fig.4.1 Tree structure of CELLA code.

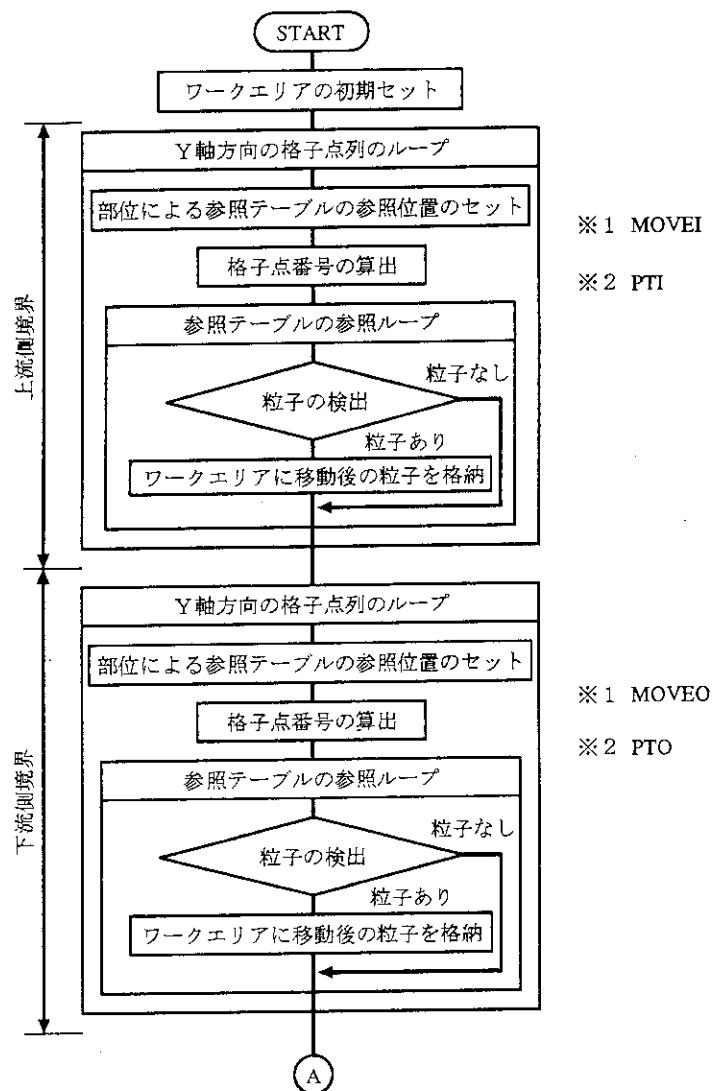


Fig.4.2 Schematic flowchart of subroutine MOVE part 1.

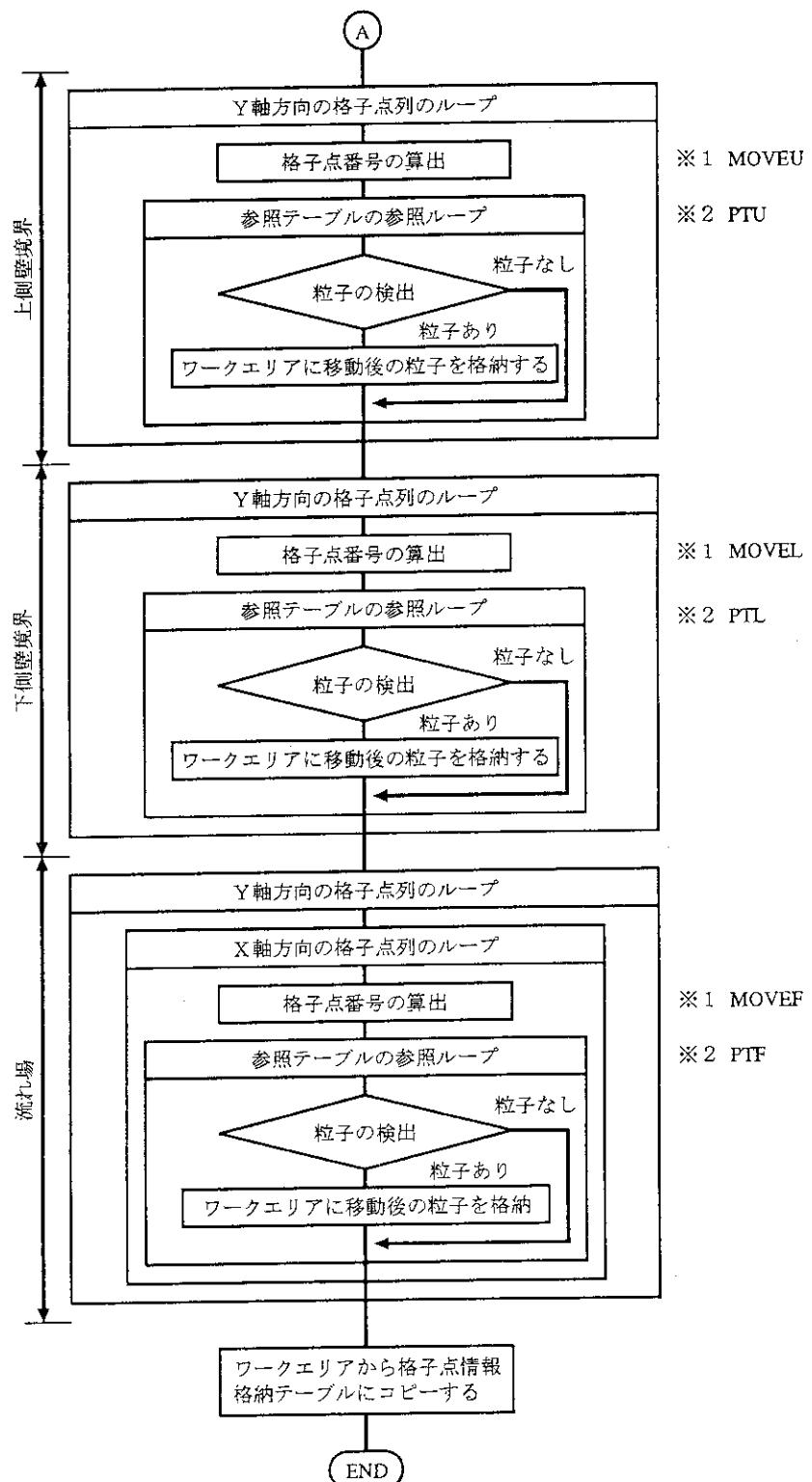


Fig.4.3 Schematic flowchart of subroutine MOVE part 2.

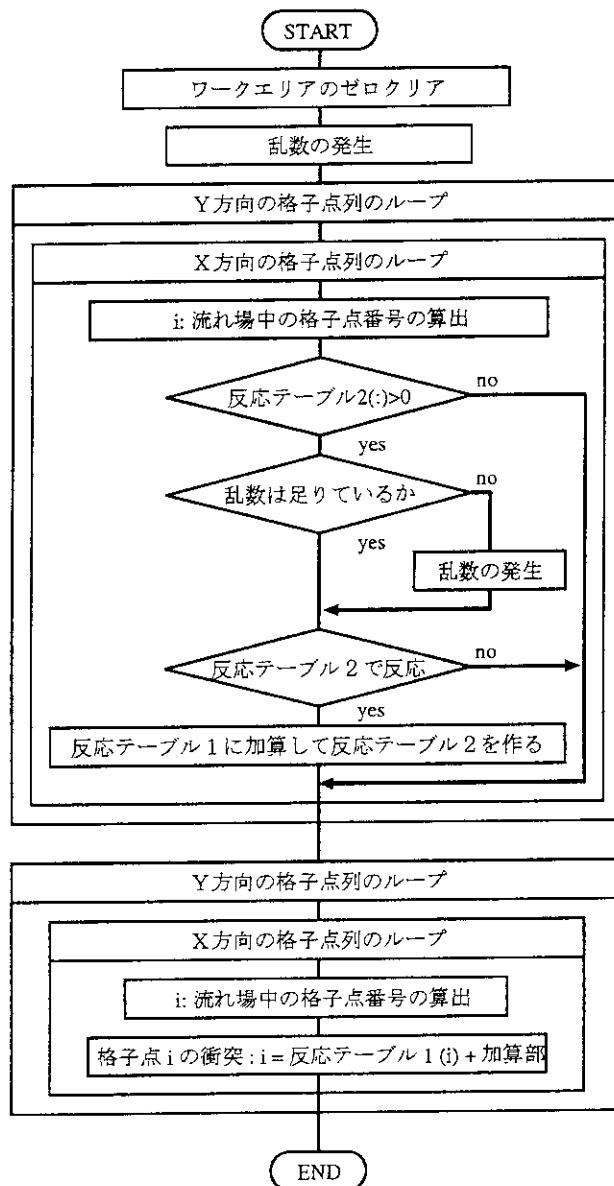


Fig.4.4 Schematic flowchart of subroutine COLLIS.

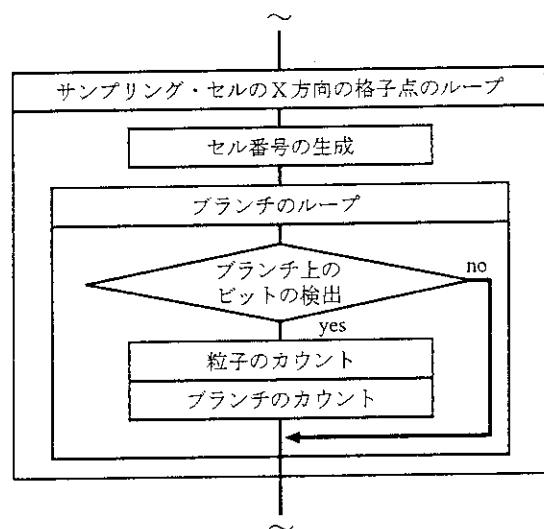
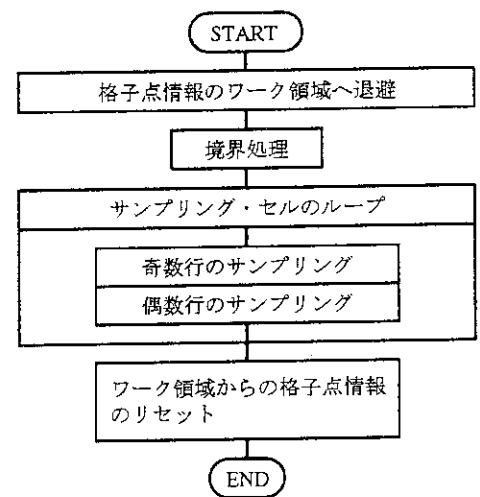


Fig.4.5 Schematic flowchart of subroutine SAMPLE.

```

SSN          VO C-----1-----2-----3-----4-----5

00001204 1-----S-----DO 550 IH=2,HEIGHT-1
00001205 1 2-----S-----DO 550 IW=2,WIDTH+(1-MOD(IH,2))-1
00001206 1 2           I=WIDTH*(IH-1)+IW+INT(REAL(IH-1)/2.)
00001207 1 2 3-----S-----DO 500 J=1,6
00001208 1 2 3 4----V----IF(IOR(GRID(I),MOVEF(J))-GRID(I).EQ.0) THEN
00001209 1 2 3 4      M           GWK(I+PTF(J))=IOR(GWK(I+PTF(J)),MOVEF(J))
00001210 1 2 3 +----V----ENDIF
00001211 1 2 +----V---500 CONTINUE
00001212 +-----S---550 CONTINUE

```

Fig.4.6 A part of subroutine MOVE.

```

SSN          VO C-----1-----2-----3-----4-----5

00001218 1-----S-----DO 500 J=1,6
00001219 1 2-----S-----DO 500 IH=2,HEIGHT-1
                  *VOCL LOOP,NOVREC
00001220 1 2 3----V----DO 550 IW=2,WIDTH+(1-MOD(IH,2))-1
00001221 1 2 3      V           I=WIDTH*(IH-1)+IW+INT(REAL(IH-1)/2.)
00001222 1 2 3 4----V----IF(IOR(GRID(I),MOVEF(J))-GRID(I).EQ.0) THEN
00001223 1 2 3 4      V           GWK(I+PTF(J))=IOR(GWK(I+PTF(J)),MOVEF(J))
00001224 1 2 3 +----V----ENDIF
00001225 1 2 +----V---550 CONTINUE
00001226 +-----S---500 CONTINUE

```

Fig.4.7 A part of modified subroutine MOVE.

```

SSN          VO C-----1-----2-----3-----4-----5

00001138           IW=1
00001139 1-----S-----DO 150 IH=1,HEIGHT
00001140 1 2-----V-----IF(MOD(IH,2).NE.0) THEN
00001141 1 2 3-----V-----IF(IH.EQ.1) THEN
00001142 1 2 3      M      JS = 3
00001143 1 2 3      S      JE = 5
00001144 1 2 +-----V-----ELSEIF(IH.EQ.HEIGHT) THEN
00001145 1 2 3      M      JS = 1
00001146 1 2 3      M      JE = 3
00001147 1 2 +-----V-----ELSE
00001148 1 2 3      S      JS = 1
00001149 1 2 3      M      JE = 5
00001150 1 2 +-----V-----ENDIF
00001151 1 +-----V-----ELSE
00001152 1 2      S      JS = 2
00001153 1 2      M      JE = 4
00001154 1 +-----V-----ENDIF
00001155 1      V      I = WIDTH*(IH-1) + IW + INT(REAL(IH-1)/2.)
00001156 1 2-----S-----DO 100 J=JS,JE
00001157 1 2 3-----V-----IF(IOR(GRID(I),MOVEI(J))-GRID(I).EQ.0) THEN
00001158 1 2 3      M      GWK(I+PTI(J)) = IOR(GWK(I+PTI(J)),MOVEI(J))
00001159 1 2 +-----V-----ENDIF
00001160 1 +-----V---100 CONTINUE
00001161 +-----V---150 CONTINUE

```

Fig.4.8 A part of subroutine MOVE (inflow section).

SSN VO C-----1-----2-----3-----4-----5

```

00001152                IW=1
                      *VOCL LOOP,NOVREC
00001153 1-----V-----DO 150 IH=1,HEIGHT
00001154 1              V            I = WIDTH*(IH-1) + IW + INT(REAL(IH-1)/2.)
00001155 1 2-----V-----IF(MOD(IH,2).NE.0) THEN
00001156 1 2 3-----V-----IF(IH.EQ.1) THEN
00001157 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(3))-GRID(I).EQ.0) THEN
00001158 1 2 3 4        V            GWK(I+PTI(3)) = IOR(GWK(I+PTI(3)),MOVEI(3))
00001159 1 2 3 +-----V-----ENDIF
00001160 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(4))-GRID(I).EQ.0) THEN
00001161 1 2 3 4        V            GWK(I+PTI(4)) = IOR(GWK(I+PTI(4)),MOVEI(4))
00001162 1 2 3 +-----V-----ENDIF
00001163 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(5))-GRID(I).EQ.0) THEN
00001164 1 2 3 4        V            GWK(I+PTI(5)) = IOR(GWK(I+PTI(5)),MOVEI(5))
00001165 1 2 3 +-----V-----ENDIF
00001166 1 2 +-----V-----ELSEIF(IH.EQ.HEIGHT) THEN
00001167 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(1))-GRID(I).EQ.0) THEN
00001168 1 2 3 4        V            GWK(I+PTI(1)) = IOR(GWK(I+PTI(1)),MOVEI(1))
00001169 1 2 3 +-----V-----ENDIF
00001170 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(2))-GRID(I).EQ.0) THEN
00001171 1 2 3 4        V            GWK(I+PTI(2)) = IOR(GWK(I+PTI(2)),MOVEI(2))
00001172 1 2 3 +-----V-----ENDIF
00001173 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(3))-GRID(I).EQ.0) THEN
00001174 1 2 3 4        V            GWK(I+PTI(3)) = IOR(GWK(I+PTI(3)),MOVEI(3))
00001175 1 2 3 +-----V-----ENDIF
00001176 1 2 +-----V-----ELSE
00001177 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(1))-GRID(I).EQ.0) THEN
00001178 1 2 3 4        V            GWK(I+PTI(1)) = IOR(GWK(I+PTI(1)),MOVEI(1))
00001179 1 2 3 +-----V-----ENDIF
00001180 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(2))-GRID(I).EQ.0) THEN
00001181 1 2 3 4        V            GWK(I+PTI(2)) = IOR(GWK(I+PTI(2)),MOVEI(2))
00001182 1 2 3 +-----V-----ENDIF
00001183 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(3))-GRID(I).EQ.0) THEN
00001184 1 2 3 4        V            GWK(I+PTI(3)) = IOR(GWK(I+PTI(3)),MOVEI(3))
00001185 1 2 3 +-----V-----ENDIF
00001186 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(4))-GRID(I).EQ.0) THEN
00001187 1 2 3 4        V            GWK(I+PTI(4)) = IOR(GWK(I+PTI(4)),MOVEI(4))
00001188 1 2 3 +-----V-----ENDIF
00001189 1 2 3 4-----V-----IF(IOR(GRID(I),MOVEI(5))-GRID(I).EQ.0) THEN
00001190 1 2 3 4        V            GWK(I+PTI(5)) = IOR(GWK(I+PTI(5)),MOVEI(5))
00001191 1 2 3 +-----V-----ENDIF
00001192 1 2 +-----V-----ENDIF
00001193 1 +-----V-----ELSE
00001194 1 2 3-----V-----IF(IOR(GRID(I),MOVEI(2))-GRID(I).EQ.0) THEN
00001195 1 2 3        V            GWK(I+PTI(2)) = IOR(GWK(I+PTI(2)),MOVEI(2))
00001196 1 2 +-----V-----ENDIF
00001197 1 2 3-----V-----IF(IOR(GRID(I),MOVEI(3))-GRID(I).EQ.0) THEN
00001198 1 2 3        V            GWK(I+PTI(3)) = IOR(GWK(I+PTI(3)),MOVEI(3))
00001199 1 2 +-----V-----ENDIF
00001200 1 2 3-----V-----IF(IOR(GRID(I),MOVEI(4))-GRID(I).EQ.0) THEN
00001201 1 2 3        V            GWK(I+PTI(4)) = IOR(GWK(I+PTI(4)),MOVEI(4))
00001202 1 2 +-----V-----ENDIF
00001203 1 +-----V-----ENDIF
00001204 +-----V---150 CONTINUE

```

Fig.4.9 A part of subroutine MOVE (inflow section).

```

SSN          VO C-----1-----2-----3-----4-----5

00000378          CALL INRGEN(2,IRMAX)

00000379          L = 0
00000380 1-----S-----DO 100 IH=2,HEIGHT-1
00000381 1 2-----S-----DO 100 IW=2,WIDTH+(1-MOD(IH,2))-1
00000382 1 2       S       I = WIDTH * (IH-1) + IW + INT(REAL(IH-1) / 2.)
00000383 1 2 3-----S-----IF(REFTL2(GRID(I)).GT.0) THEN
00000384 1 2 3       S       L = L + 1
00000385 1 2 3 4-----S-----IF(L.GT.IRMAX) THEN
00000386 1 2 3 4       S       CALL INRGEN(2,IRMAX)
00000387 1 2 3 4       S       L = 1
00000388 1 2 3 +-----S-----ENDIF
1 2 3
00000389 1 2 3 4-----S-----IF(INR(L).EQ.2) THEN
1 2 3 4       S       GWK(I) = REFTL2(GRID(I)) - REFTL1(GRID(I))
1 2 3 +-----S-----ENDIF
1 2 3       S
00000390 1 2 +-----S-----ENDIF
00000391 +-----S---100 CONTINUE

```

Fig.4.10 A part of subroutine COLLIS.

```

SSN          VO C-----1-----2-----3-----4-----5

00000381           CALL INRGEN(2,IRMAX)

00000382           L=0
00000383 1-----S-----DO 100 IH=2,HEIGHT-1
00000384 1           S      NRANSU = 0
00000385 1 2-----V-----DO 110 IW=2,WIDTH+(1-MOD(IH,2))-1
00000386 1 2           V      I=WIDTH*(IH-1)+IW+INT(REAL(IH-1)/2.)
00000387 1 2 3-----V-----IF(REFTL2(GRID(I)).GT.0) THEN
00000388 1 2 3           V      IFLAG(IW) = I
00000389 1 2 3           V      NRANSU = NRANSU + 1
00000390 1 2 +-----V-----ELSE
00000391 1 2 3           V      IFLAG(IW) = 0
00000392 1 2 +-----V-----ENDIF
00000393 1 +-----V---110   CONTINUE
00000394 1           S-----IF( L+NRANSU .GT. IRMAX ) THEN
00000395 1 2           S      CALL INRGEN(2,IRMAX)
00000396 1 2           S      L = 1
00000397 1 +-----S-----ENDIF
00000398 1 2-----V-----DO 120 IW=2,WIDTH+(1-MOD(IH,2))-1
00000399 1 2 3-----V-----IF( IFLAG(IW) .NE. 0 ) THEN
00000400 1 2 3           V      I = IFLAG(IW)
00000401 1 2 3           V      L = L + 1
00000402 1 2 3 4-----V-----IF(INR(L).EQ.2) THEN
00000403 1 2 3 4           V      GWK(I) = REFTL2(GRID(I))
00000404 1 2 3 +-----V-----ENDIF
00000405 1 2 +-----V-----ENDIF
00000406 1 +-----V---120   CONTINUE
00000407 +-----S---100   CONTINUE

```

Fig.4.11 A part of modified subroutine COLLIS.

SSN	VO C-----1-----2-----3-----4-----5
00001479	1-----S-----DO 1000 IY=1,SDIVY
00001480	1 2-----S-----DO 1000 IX=1,SDIVX
00001481	1 2 S IS = (IY-1)*SDIVX + IX
00001482	1 2 S IO = WIDTH*(IY-1)*(NSCY-1) + NSCX*(IX-1)
	1 2 * + INT(REAL((IY-1)*(NSCY-1))/2.) + 1
00001483	1 2 3-----S-----DO 500 K=1,NSCY-1,2
	1 2 3
00001484	1 2 3 M I = (IO+(K-1)*WIDTH+INT(REAL(K-1)/2.))-1
	1 2 3
00001485	1 2 3 4-----DO 150 L=1,NSCX
00001486	1 2 3 4 I = I + 1
00001487	1 2 3 4 5-----V-----DO 100 M=1,7
00001488	1 2 3 4 5 6-----V-----IF(IOR(GRID(I),BITCHK(M))-GRID(I).EQ.0) THEN
00001489	1 2 3 4 5 6 V SPAR(IS) = SPAR(IS) + 1.
00001490	1 2 3 4 5 6 V SVEL(M,IS) = SVEL(M,IS) + 1.
00001491	1 2 3 4 5 +---V-----ENDIF
00001492	1 2 3 4 +---V---100 CONTINUE
	1 2 3 4
00001493	1 2 3 +---150 CONTINUE
	1 2 3
00001494	1 2 3 M I=(IO+K*WIDTH+INT(REAL(K)/2.))-1
	1 2 3
00001495	1 2 3 4-----DO 250 L=1,NSCX+1
00001496	1 2 3 4 I = I + 1
00001497	1 2 3 4 5-----V-----DO 200 M=1,7
00001498	1 2 3 4 5 6-----V-----IF(IOR(GRID(I),BITCHK(M))-GRID(I).EQ.0) THEN
00001499	1 2 3 4 5 6 V SPAR(IS) = SPAR(IS) + 1.
00001500	1 2 3 4 5 6 V SVEL(M,IS) = SVEL(M,IS) + 1.
00001501	1 2 3 4 5 +---V-----ENDIF
00001502	1 2 3 4 +---V---200 CONTINUE
00001503	1 2 3 +---250 CONTINUE
00001504	1 2 +---V---500 CONTINUE
	1 2
00001505	1 2 S I=(IO+(NSCY-1)*WIDTH+INT(REAL(NSCY-1)/2.))-1
	1 2
00001506	1 2 3-----DO 650 L=1,NSCX
00001507	1 2 3 I = I + 1
00001508	1 2 3 4-----V-----DO 600 M=1,7
00001509	1 2 3 4 5-----V-----IF(IOR(GRID(I),BITCHK(M))-GRID(I).EQ.0) THEN
00001510	1 2 3 4 5 V SPAR(IS) = SPAR(IS) + 1.
00001511	1 2 3 4 5 V SVEL(M,IS) = SVEL(M,IS) + 1.
00001512	1 2 3 4 +---V-----ENDIF
00001513	1 2 3 +---V---600 CONTINUE
	1 2 3
	1 2 +---650 CONTINUE
00061440	+---S---1000 CONTINUE

Fig.4.12 A part of subroutine SAMPLE.

5. GGR の高速化

黒鉛型結晶における照射損傷をシミュレーションする分子動力学コード GGR [3] の高速化を行った。GGR コードでは、黒鉛型結晶に特徴的な共有結合を考慮し、原子間相互作用として、中心力の他に非中心力を取り入れることによって、より現実的な結晶構造におけるシミュレーションが可能である。しかし、GGR コードはベクトル計算機向きにコーディングされていないため、ベクトル計算機による実行時間の短縮は得られない。高速化にあたりベクトル化を試みたが、GGR コードはベクトル計算に不向きなアルゴリズムのため、一部のサブルーチンのベクトル化にとどまり、このベクトル化した部分のコストが低いためにベクトル化自身による効果は少ない。しかし、オリジナル版では FORTRAN77(V10)仕様でコンパイラオプション AUTODBL(DBLPAD) を使用していたが、今回の改良で、このオプションを指定せずに倍精度計算を可能としたために、高速化がなされた。この結果ベクトル化版の VP2600/10 (以下 VP2600) 上のベクトル実行時間は、オリジナルの AUTODBL 使用版の M780/20 (以下 M780) の実行に比べて、5.6 倍、VP2600 上の AUTODBL を使用しない倍精度版のスカラ実行に比べて 1.23 倍という倍率が出た。

5.1 コード概要

GGR コードは黒鉛型結晶構造を持つ物質の照射損傷をシミュレーションするための分子動力学コードであり、PART1, PART2, および PART3 からなっている。これらの区分や流れはダイヤモンド型を扱う DGR と基本的に同じである。しかし、ダイヤモンド型結晶では直交座標系が基本であったが、黒鉛型結晶では斜交座標系を基本としている。これは、直交座標系を基本とすると結晶の対称性を利用できないためである。粒子の移動を記述するには直交座標系が適しているが、相互関係を検索するには、結晶軸に沿って斜交座標系をとった方が便利である。このために、両者をそれぞれ使い分けている。

PART1 では、結晶とその初期条件の設定、原子間力やポテンシャル関数のテーブル化、および相互作用する隣接原子を検索するテーブルの設定等を行う。PART1 の流れを Fig. 5.1 に示す。PART2 では PART1 で設定されたデータファイルを読み込んで、結晶を構成する各粒子に対する運動方程式を解き、種々のエネルギーについて計算する。中心力の他に、共有結合における変角力を模擬するための非中心力作用を扱っている点が特徴である。PART2 の流れを Fig. 5.2 に示す。PART3 では、PART2 で出力された計算結果を図形出力する。

5.2 FORTRAN77EXへの対応

前節のとおり、GGR コードは PART1 から PART3 までの 3 段階から成り立っている。これらのコードは FORTRAN-HE コンパイラの文法でコーディングされており、現在使用されている FORTRAN77EX コンパイラではコンパイルできない。このため FORTRAN77EX コンパイラに対応させるために変更を行った。

5.2.1 組込み関数の変更

GGR コードでは、いくつかの組込み関数を参照しているが、その中に FORTRAN77EX の組み込み関数と名前が異なるものがある。このため組み込み関数名を変更した。修正したサブルーチンは GR243(PART2) である。

変更した組み込み関数

DARCOS → DACOS

5.2.2 プログラム単位に関する宣言文の変更

メインプログラムに名前を付けるとき、FORTRAN-HE では ELEMENT 文を使用する。しかし、JIS 規格の FORTRAN 文法では、PROGRAM 文を用いると規定されている。このため ELEMENT 文から PROGRAM 文に変更する。変更を行ったサブルーチンは GR100(PART1), GR200(PART2), GR300(PART3) である。

宣言の変更例

ELEMENT GR100 → PROGRAM GR100

5.2.3 倍精度化

GGR コードの FORTRAN-HE 版では、コンパイル時にコンパイルオプション 'AUTODBL(DBLPAD)' を指定して倍精度化を行っていた。FORTRAN77EX にも AUTODBL や DOUBLE のオプションはあるが、今回の改良でコード自体を倍精度化した。

倍精度化の変更では変換ツール STREAM77 を使用して、各サブルーチンに IMPLICIT REAL*8(A-H,O-Z) を挿入することと、実定数の倍精度化を行った。この変更は PART1, PART2, PART3 のすべてのサブルーチンに対して行われた。

倍精度化の変更例

SUBROUTINE ABC(XX,YY)	SUBROUTINE ABC(XX,YY)
YY = XX * 3.14	→ IMPLICIT REAL*8 (A-H,O-Z)
RETURN	YY = XX * 3.14D0
END	RETURN
	END

5.3 ベクトル化

GGR コードを構成する PART1, PART2, および PART3 で、今回の入力データを用いた実際の CPU 時間はそれぞれ 0.17 秒, 9 分 17.75 秒, 1.95 秒 (全て VP2600 上) である。したがって計算時間が最も長い PART2 をベクトル化した。PART2 は結晶を構成する原子の運動方程式を解き、各原子の位置を時間を追って計算すると共に、全エネルギー等も計算する。このため、粒子の数を多くしたり、解析時間を長くすると、計算時間はさらに長くなる。

PART2 の処理を見ると、各粒子については原則的に独立に計算できる構造になっているため、ベクトル化の方針としては粒子数に関するベクトルループが出来るように処理を変更する。この結果、粒子数が多くなるほどベクトル長が長くなるため、より高い速度向上率が期待される。

5.3.1 動的挙動解析

PART2 の実際のコスト分布を調査するため、動的挙動解析ツール ANALYZER [11] を用いた解析を行った。その結果を Table 5.1 に示す。この表によるとサブルーチン GR241, GR247A, GR247, GR245, GR246 の 5 つで全コストの 90 % を越え、また実行回数も非常に多いので、これらをベクトル化の対象とする。

PART2 の計算の流れは Fig. 5.2 に示した。この中で、サブルーチン GR230 から GR260 に至る大粒度のループが存在するが、これは時間のループのためベクトル処理できない。GR240 の中には粒子のループがあり、このループ内で各粒子に対する運動方程式を解く。このループをベクトル処理する。今回使用した入力データでは粒子数 938 であり、十分に長いベクトル長が得られるため、高速化が期待できる。将来的には、数万個の粒子を用いたシミュレーションを行う予定なので、さらに高い速度向上率が得られることになる。またこのループ内では GR241, さらには GR241 からは GR245, GR247, GR247A 等いくつかのサブルーチンを呼び出しているため、呼び出しにかかるオーバーヘッドが大きい。これらの調査結果を基にベクトル化作業を進めた。

5.3.2 ベクトル化のための変更

GR240 は各粒子についての計算を行う部分であり、PART2 全体の処理の中で中心的な部分である。オリジナル版では、PART1 の出力結果より、各粒子の初期条件（座標等）を入力し、それを元に、粒子の相互作用による各粒子に働く力と表面力による各原子の変位量を計算する。すべての粒子について計算が終わった後、粒子の新しい位置（座標）を計算し、再び物性値の計算を進めていく。

近接した粒子との相互作用は、「箱検索」という方法を用いて計算される。これは粒子をいくつかの箱に入れて、インデックスで指示した箱に対して、その箱に属する粒子との相互作用を計算するものである。GGR では 1 つの箱の中に 4 つまでの粒子が入る。一つの粒子に対する相互作用の計算は、DGR コードと同様、その粒子含む箱とその回りに近接する箱 26 個（計 27 個、ただし境界部分は境界条件が設定されるので、少なくなる）に含まれる粒子について行なわれる。また、箱に存在する粒子は時間がたつにつれて他の箱に移動する場合がある。

コード上ではサブルーチン GR240 で粒子に関するループがあり、その中で、GR241 を呼び出している。GR241 では加速度や位置エネルギー等を求める。GR241 のベクトル化では GR240 にある粒子ループを GR241 に引き込み、このループでベクトル化することになる。

5.3.2.1 サブルーチン GR240

GR240 は粒子ループ (DO 5000) 内の先頭で GR241 の呼び出しがあり、引数として粒子番号を渡している。以下の処理では、GR241 で計算された種々のパラメータを使って、位置エネルギーや加速度の計算を行っている。GR241 の中にベクトル化するために粒子ループを引き込ん

だ. GR241 の呼び出し時の引数に GR246 で使用されるフラグと粒子番号を指定していたが, 粒子ループを GR241 に引き込んだために, 引数を省略した. これらの変数はもともと両方のサブルーチンで COMMON 宣言してある. 加速度 AC は全ての粒子について一度に GR241 で計算するが, GR241 の呼び出し以降の処理で AC を使用するので, 全ての粒子の加速度を保存しておくために, 配列 AC の次元を上げた ($AC(3) \rightarrow AC(4000,3)$).

呼び出し以降の処理では, 粒子ループについて回帰計算などのベクトル処理不可能な部分がないので, ベクトル処理が可能だが, コーディングが複雑なためベクトル処理が一部できない. できない箇所は弾性エネルギー ENGPO, および損失エネルギー ENGL を求める部分である. それぞれ 2 カ所現れる. そこで, 後に現れる ENGPO, および ENGL の名前を他の一時的な作業用変数に名前を変え, DO ループの直後でそれぞれ ENGPO, ENGL に足し込むようにした. これ以外では, 小さな DO ループがいくつか存在するが, すべてコンパイラによりアンローリングされ, 粒子ループでベクトル処理が可能となった. オリジナルのこのループを Fig. 5.3 に, ベクトル化版を Fig. 5.4 に示す.

5.3.2.2 サブルーチン GR241, GR243, GR244, GR245, GR246, GR247, GR247A

GR241 の処理の概要を Fig. 5.5 に示す. オリジナル版では, 親ルーチンである GR240 の粒子ループの中で呼ばれているが, ベクトル化のために粒子ループをこのルーチンに引き込む. ベクトル化するために GR241 を大きく 3 つのブロックに分けた. 初期値設定部分と, 加速度, および角度計算部分が 2 カ所である.

第 1 ブロックの初期値設定部分では, GR241 内で使用される角度, 距離等が入る配列 (RNF, R 他) のゼロクリアをしており, GR241 が呼び出されるたびに実行される. また, ここでは箱がある位置における粒子数の設定 (配列 NVAL) も行っている. これは箱の位置によって (境界条件があるために) 含まれる粒子数が異なるので, この定義を行う. この初期値設定部分は粒子ループでベクトル処理される.

第 2 ブロックの加速度, および角度計算部分はオリジナル版では LL=1 から DO 1900 の直前までである. この部分はベクトル化版の DO 4300 の粒子ループに相当する. ループの先頭に AC の定義部分を, 初期値設定部分より, プログラムの構造上移動する必要がある. このループはベクトル化不可能である. 原因はベクトル処理を行うと, 回帰計算が発生するからである.

```

DO 4300 NO = 1, N
    AC(NO,1) = ZZ1(NO)
    AC(NO,2) = ZZ2(NO)
    AC(NO,3) = ZZ3(NO)
    .
    .
    I4 = IBOXW(4)
    .
    .
    AC(NO,1) = AC(NO,1) + ACC*XW/WRK

```

```

AC(NO,2) = AC(NO,2) + ACC*YW/WRK
AC(NO,3) = AC(NO,3) + ACC*ZW/WRK
ZZ1(I4) = ZZ1(I4) - ACC*XW/WRK
ZZ2(I4) = ZZ2(I4) - ACC*YW/WRK
ZZ3(I4) = ZZ3(I4) - ACC*WW/WRK
.
.
.
```

4300 CONTINUE

ここで加速度 AC は、配列 ZZ1 ~ ZZ3 によって値を定義している。しかし、ZZ1 等は添字に I4 が使用されているので、(変数 NO とのかねあいで) 回帰計算が発生する。そのため、このループはベクトル化できない。しかし、サブルーチンの呼び出しのオーバーヘッドをなくすため IC, サブルーチン GR247, および GR247A をこのルーチンで展開した。

第3ブロックの加速度、および角度計算部分は、粒子ループでベクトル化される。ただしベクトル化を行うために、いくつかの改良を施した。この粒子ループの中でいくつかの小さいループが存在するので、すべて展開した。また、このサブルーチンから GR245 を呼び出し、さらに GR245 から GR243, GR244, GR246 を呼び出している。CALL 文はベクトル化非対象であるため、GR245, GR244, GR246 を実際に上位展開を行い、GR243, GR246 はコンパイル・オプションに INLINE 展開を行う最適化レベル OPT(F) を指定することで、上位展開を行った。ただし GR246 に関しては WRITE 文を含むので、この WRITE 文を GR240 に移動した。しかし、これらの改良を行ってもすべてがベクトル化されない。これは、コーディングが複雑すぎて、コンバイラが解析不可能であるからである。そのため、このサブルーチン内のみ使用される作業配列 (PHAI1, THRO 等) はすべて配列をやめ、変数化を行った。これに伴い、コンバイラによる上位展開を行う GR243 の引数も作業配列を指定しているので、変更した変数に置き換えた。これで、このループのベクトル処理が可能となった。オリジナルの GR241 を Fig. 5.6 IC, ベクトル化版の GR241 を Fig. 5.7 IC に示す。

5.4 ベクトル化の効果

ここでは、ベクトル化版の計算結果、および効果について述べる。

5.4.1 計算結果

作成したベクトル化版を用いて計算を行った。得られた計算結果はオリジナルとすべての桁で一致した。

5.4.2 ベクトル化の効果

ベクトル化版に対し ANALYZER を用いてコスト分布を調査した。その結果を Table 5.2 IC 示す。GR241 では下位ルーチンの GR243, GR246 等を引き込んだので、コスト比率が上昇している。しかし、GR241 の一部はベクトル処理可能である。

また、2つの入力データを用いてベクトル化版の速度向上率を測定した。この結果を Table 5.3 に示す。コンパイラはスカラ計算は FORTRAN77EX(V12) の OPT(E)、ベクトル計算は FORTRAN77EX/VP(V12) の OPT(F) で行った。この表によれば、入力データ TYPE1 で VP2600 上でオリジナルスカラに対してベクトル化版は約 1.23 倍、M780 上のオリジナル版との比較では 2.10 倍の倍率が出た。TYPE2 でも同様な結果となった。倍率が低い原因是、1 番コストの高いサブルーチン GR241 の中の、コストの高い角度・加速度計算部分がベクトル化できないことがある。

5.5 実行環境

ここではベクトル化版の GGR コードを実行する際の実行環境について述べる。

5.5.1 データセット関係

GGR コードのオリジナル版は FORTRAN-HE による単精度のコードであった。今回の作業によって、FORTRAN77EX 仕様の倍精度コードを作成し、さらにベクトル化版を作成した。これらのデータセットを Table 5.4 に示す。

5.5.2 ロードモジュール作成、および実行方法

5.5.2.1 ロードモジュール作成

Fig. 5.8 に GGR ベクトル化版ロードモジュールを作成する JCL を示す。ベクトル化版のロードモジュールを作成する場合サブルーチン GR243, GR246 を内部展開させる必要があるため、コンパイルオプションに OPT(F) を指定する必要がある。

5.5.2.2 実行

実行方法は VP で実行するための CLASS 指定することを除いて、今までの実行とほとんど変更はない。実行 JCL を Fig. 5.9 に示す。

5.5.2.3 実行時の注意

解析の対象となるデータで、粒子数を多くする場合はベクトル化作業領域を大きくする必要がある。オリジナルでは、解析可能な最大粒子数は 4000 であり、コード内で使用されている配列もすべて大きさ 4000 で宣言されている。粒子数を多くする場合は、これらの配列を変更するとともに、ベクトル化作業用の配列も変更する必要がある。Table 5.5 にベクトル化版に追加された配列を示す。

5.6 まとめ

今回 GGR コードのベクトル化を行ったが、純粋なベクトル化で高速化はできなかった。原因是、一番コストのかかる計算部分にベクトル処理不可能な回帰計算があるためである。しかし、AUTODBL(DBLPAD) によるコードの倍精度化から、コード自身を実際に倍精度化することによって、約 2.6 倍の高速化が達成された。AUTODBL を使用しないとなぜ、このような高速化がなされるかは現在調査中であるが、この改良だけでも、このコードは使いやすくなつたと思われる。

Table 5.1 Dynamic Behavior of the Original Version of GGR (PART2) code

<< ANALYZER TOTAL LIST : COST TABLE >>

Table 5.2 Dynamic Behavior of the VEctor Version of GGR (PART2) code

```
<< ANALYZER TOTAL_LIST : COST_TABLE >>
```

RANK	NAME	EX-COUNT	CPU-AVG	VU-AVG	CPU-TIME	VU-TIME	χ	V-COST	S-COST	χ	OVERHD
1	GR24.1	201						6331E10	6569E10	97.3	0
2	GR24.0	201						4803057	1026E09	1.5	0
3	GR25.0	201						12010169	0.1	23051584	0.3
4	GR20.0	1						10265810	0.1	45975276	0.4
5	GR23.0	201						10207785	0.1	10207785	0.1
6	GR26.0	21						399	0.0	399	0.0
7	GR21.0	1						315	0.0	315	0.0
8	GR21.0	1						254	0.0	658	0.0
9	GR22.0	0						0	0	0	0
10	GR24.3	0						0	0	0	0
11	GR24.4	0						0	0	0	0
12	GR24.5	0						0	0	0	0
13	GR24.6	0						0	0	0	0
14	GR24.7	0						0	0	0	0
15	GR24.7A	0						0	0	0	0
(TOTAL)											
								6408E10	100.0	6751E10	100.0

Table 5.3 CPU Time Table of the Original and Vectorized Versions

項目		入力データ	TYPE1	TYPE2
CPU 時間	オリジナル 単精度(M780) * 1	5m48s48	14m25s09	
	オリジナル 単精度(VP2600) * 1	3m17s98	8m13s49	
	オリジナル(M780)	2m11s34	5m24s82	
	オリジナル(VP2600)	1m16s85	3m11s94	
	VP版スカラモード	1m07s43	2m47s86	
	VP版VPモード (VU時間)	1m02s46 (0s46)	2m35s29 (1s15)	
使用領域 (KB)	オリジナル 単精度(M780) * 1	1836	1836	
	オリジナル 単精度(VP2600) * 1	2028	2028	
	オリジナル(M780)	1704	1704	
	オリジナル(VP2600)	1896	1896	
	VP版スカラモード	2504	2504	
	VP版VPモード	2792	2804	
I/O 回数	オリジナル 単精度(M780) * 1	395	835	
	オリジナル 単精度(VP2600) * 1	394	834	
	オリジナル(M780)	228	450	
	オリジナル(VP2600)	227	449	
	VP版スカラモード	227	449	
	VP版VPモード	227	449	
倍率	オリジナル 単精度(M780)/VP 版VPモード	5.57	5.57	
	オリジナル 单精度(VP2600)/VP 版VPモード	3.17	3.17	
	オリジナル(M780)/VP版VPモード	2.10	2.10	
	オリジナル(VP2600)/VP版VPモード	1.23	1.23	
	VP版スカラモード/VP 版VPモード	1.07	1.08	
	ベクトル化率(%)	8.1	8.2	

注1：コンパイルオプションにAUTODBL(DBLPAD)を指定している。

Table 5.4 Datasets of GGR Source and JCL

バージョン	データセット名		メンバ名	補足
オリジナル版	PART1	J1476.ZGR11.FORT77		FORTRAN-HE仕様 単精度版
	PART2	J1476.ZGR12.FORT77		
	PART3	J1476.ZGR13.FORT77		
オリジナル版	PART1	J0001.ZGR11D.FORT77		FORTRAN77EX 仕様 倍精度版
	PART2	J0001.ZGR12D.FORT77		
	PART3	J0001.ZGR13D.FORT77		
ベクトル化版	PART2	J0001.ZGR12DVP.FORT77		FORTRAN77EX 倍精度
J C L	PART1	J0001.ZGRVP.CNTL	PART1	翻訳から実行まで
	PART2		FORTCL	ロードモジュール作成
	PART3		GO	実行
			PART3	翻訳から実行まで

Table 5.5 Added or Modified Arrays

サブルーチン名	ベクトル化版での宣言	旧版での宣言
G R 2 4 6	AC(4000, 3)	AC(3)
G R 2 4 7	AC(4000, 3)	AC(3)
G R 2 4 7 A	AC(4000, 3)	AC(3)
G R 2 D 4 0	AC(4000, 3)	AC(3)
G R 2 D 4 1	AC(4000, 3) I2(4000) J1(4000) NVAL(4000) ICOMB(4000) NF(4000, 3) RNF(4000, 3) R(4000, 3, 3)	AC(3) なし なし なし なし NP(3) RNF(3) R(3, 3)

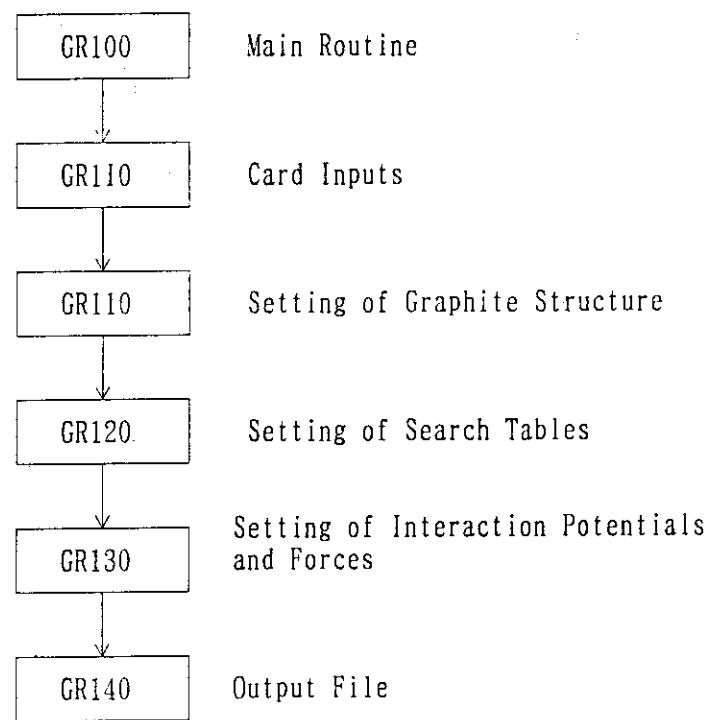


Fig.5.1 Flow Chart of the PART1 of the GGR Code

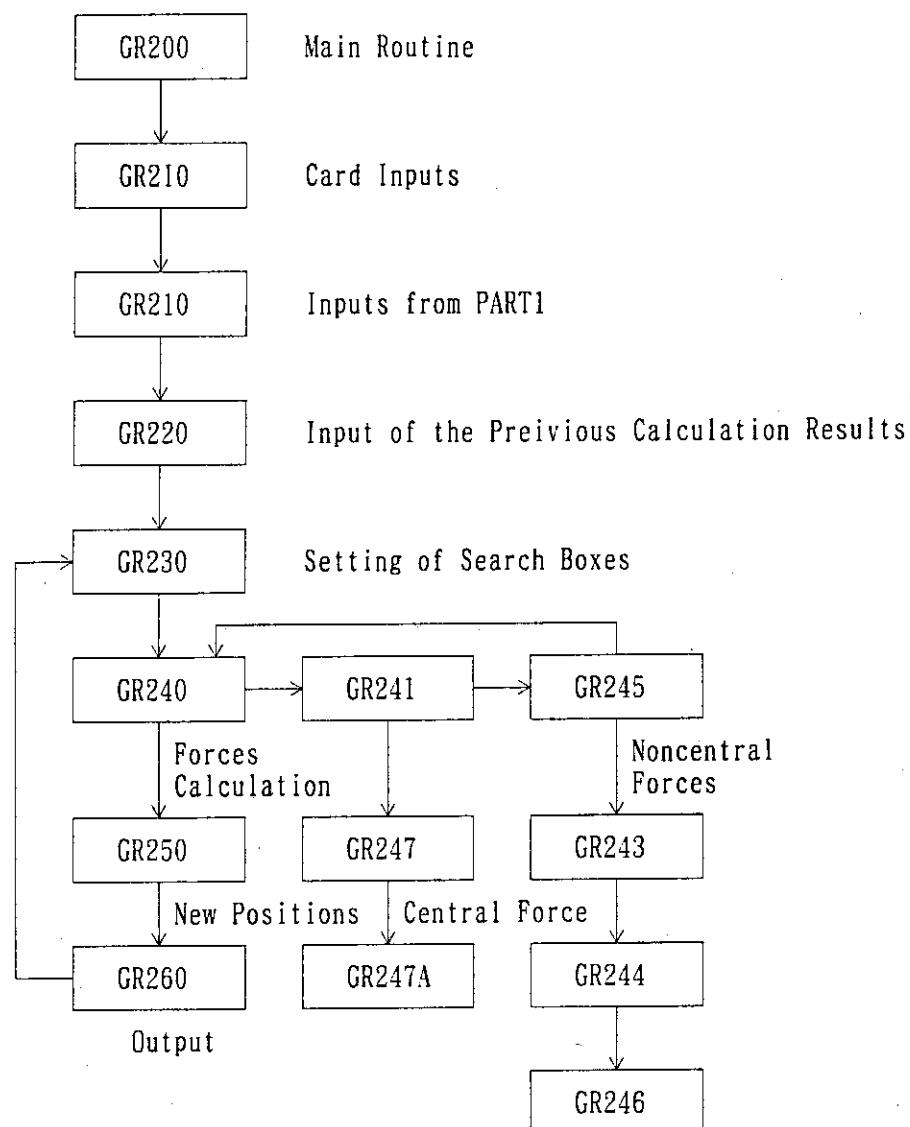


Fig.5.2 Flow Chart of the PART2 of the GGR Code

JAERI-Data/Code 94-021

Fig. 5.3 Subroutine GR240

```

★ GR240 〈 4 〉

C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
      FILE=J9127.ZGR12D.FORT77(GR240)
C
      2000 L= IBND(I)
      IEO=1
      IF(MOD(NI3(I),2).EQ.0) IEO=2
      DO 2500 J=1,3
      X(I,J)=TB(I,J)
      VX(I,J)=TB(I,J)
      ACC(J)=0.0
      ACC(I,J)=0.0
      DXYZ= X(I,J)-TB(I,J)
      ENGL=ENGSP-A(CL,J,IEO)*DXYZ-0.5DO*B(CL,J,IEO)*DXYZ**2
      ENGL=ENGL-C(CL,J,IEO)*(VX(I,J)-X(I,J))*2
      ACC(I,J)=AC(CL,J,IEO)+B(CL,J,IEO)*DXYZ+C(CL,J,IEO)*(VX(I,J)-X(I,J))
      ACC(J)= AC(J) + ACC(I,J)
      C
      IF(C11FLG(7).NE.1)          GO TO 3200
      IF(C1240.NE.1)              GO TO 3200
      SAVE EXTENAL FORCE.

      C-----DO 3100 J=1,3
      -3100 FX1(J,I)=AC(J)
      C
      3200 CONTINUE
      C
      FIXING FORCE.

      C-----DO 3250 J=1,3
      -3250 AC(J)=AC(J)-FX1(J,I)
      C
      3000 CONTINUE
      C
      3400 CONTINUE
      C
      **** PARAMETER OF MOTION ***
      C
      CXXXXXXXXXXXXXX
      * WRITE(6,6200) 1, DDT, AC, (VX(I,J),J=1,3)
      * 6200 FORMAT(1H '***GR240**',2X,'I=',13,2X,'DDT=',F8.5,2X,'AC='
      * ,3(1PD15.7),2X,'VX=',3(1PD15.7))
      CXXXXXXXXXXXXXX
      C-----DO 3500 J=1,3
      -3500 VX(I,J)= 2.0DO*(AC(J)*DDT+VX(I,J)) - X(I,J)
      C
      3500 CONTINUE
      CXXXXXXXXXXXXXX
      * WRITE(6,6300) X(35,1), (X(I,J),J=1,3), (VX(I,J),J=1,3)
      * 6300 FORMAT(1H '***GR240**',1PD15.7,*X =, 3(1PD15.7),*VX =
      * ,3(1PD15.7))
      CXXXXXXXXXXXXXX
      C-----5000 CONTINUE
      C
      1240=0
      C
      RETURN
      END

```

Fig.5.3 Subroutine GR240 (continued)

```

      **** GR240 ****
      SSN=0001
      SSN=0002
      SSN=0003
      SSN=0004
      SSN=0005
      SSN=0006
      SSN=0007
      SSN=0008
      SSN=0009
      SSN=0010
      SSN=0011
      SSN=0012
      SSN=0013
      SSN=0014
      SSN=0015
      SSN=0016
      SSN=0017
      SSN=0018
      SSN=0019
      SSN=0020
      SSN=0021
      SSN=0022
      SSN=0023
      SSN=0024
      SSN=0025
      SSN=0026
      SSN=0027
      SSN=0028
      SSN=0029
      SSN=0030
      SSN=0031
      SSN=0032
      SSN=0033
      SSN=0034
      SSN=0035
      SSN=0036
      SSN=0037
      SSN=0038
      SSN=0039
      SSN=0040
      SSN=0041
      SSN=0042
      SSN=0043
      SSN=0044
      SSN=0045
      SSN=0046
      SSN=0047
      SSN=0048
      SSN=0049
      SSN=0050
      SSN=0051
      SSN=0052
      SSN=0053
      SSN=0054
      SSN=0055
      SSN=0056
      SSN=0057
      SSN=0058
      SSN=0059
      SSN=0060
      SSN=0061
      SSN=0062
      SSN=0063
      SSN=0064
      SSN=0065
      SSN=0066
      SSN=0067
      SSN=0068
      SSN=0069
      SSN=0070
      SSN=0071
      SSN=0072
      SSN=0073
      SSN=0074
      SSN=0075
      SSN=0076
      SSN=0077
      SSN=0078
      SSN=0079
      SSN=0080
      SSN=0081
      SSN=0082
      SSN=0083
      SSN=0084
      SSN=0085
      SSN=0086
      SSN=0087
      SSN=0088
      SSN=0089
      FILE=j9127.ZGR12DVP.FORT77(GR240)          FDRTRAN 77/V/P V12L10
      C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      C-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*-----*
      C-----SUBROUTINE GR240(ACC)
      C-----IMPLICIT REAL*8 (A-H,O-Z)
      C-----          GR240    -- S --
      C
      C 1. FUNCTION
      C (1) CALCULATE DISSIPATION ENERGY, SPRING ENERGY, PRESSURE ENERGY
      C     ACCELERATION,PARAMETER OF MOTION
      C
      C-----* MAIN LOOP *
      C   TN  DO 5000 I=1,N
      C-----* GET ACCELERATION AND POTENTIAL
      C-----* ENERGY
      C-----* DIVIDE CHECK PARAMETER IN GR246
      C   TN  CALL GR241(I240,I)
      C   TN  CALL GR241
      C-----IF(IDC .EQ.1) THEN
      C-----  WRITE(6,600)
      C-----  600 FORMAT('*** GR246 ***')
      C-----END IF
      C-----*****+
      C
      C   WENGSP = 0
      C   WENGL = 0
      C-----DO 5000 I=1,N
      C-----  IF( IBND(I) ) 1000, 3400, 2000
      C-----  C
      C-----  C
      C-----  V 1000 L=-IBND(I)
      C-----  V 1E0*2
      C-----  V 1F(MOD(N13(I),2).EQ.0) 1E0=1
      C-----  V3-----1500 J=1,3
      C-----  V3-----DXYZ=X(1,J)-T(B(1,J)
      C-----  V3-----ENGSP=ENGSP-A2(L,J,IEO)*DXYZ-O*DXYZ**2
      C-----  V3-----ENGL=C2(L,J,IEO)*(UX(1,J)-X(1,J))*2
      C-----  V3-----ACC(I,J)=A2(L,J,IEO)+B2(L,J,IEO)*DXYZ+C2(L,J,IEO)*(VX(1,J)-X(1,J))
      C-----  C,500 ACC(J)=ACC(I,J)+ACC(I,J)
      C-----  V3-1500 ACC(I,J)= ACC(I,J) + ACC(I,J)
      C
      C-----00001370
      C-----00001380
      C-----00001390
      C-----00001400
      C-----00001410
      C-----00001420
      C-----00001430
      C-----00001440
      C-----00001450
      C-----00001460
      C-----00001470
      C-----00001480
      C-----00001490
      C-----00001500
      C-----00001510
      C-----00001520
      C-----00001530
      C-----00001540
      C-----00001550
      C-----00001560

```

Fig. 5.4 Modified Subroutine GR240

```

★ GR240 ( 4 )

FILE=J9127.ZGR12DVP.FORT77<GR240>
C-----1-----2-----3-----4-----5-----6-----7-----8
      IF(LIFLG(7).NE.-1)          GO TO 1510
      IF(L1240.NE.-1)          GO TO 1510
      C
      V3----DO 1505 J=1,3
      C1505 FX2(J,I)=AC(J,J)
      C1505 FX2(J,I)=AC(I,J)
      C
      1510 CONTINUE
      C
      C
      FIXING FORCE.
      C
      V3----DO 1520 J=1,3
      C1520 AC(J)=AC(J)-FX2(J,I)
      C1520 AC(I,J)=AC(I,J)-FX2(J,I)
      C
      GO TO 3000
      C
      C
      FIRST
      V 2000 L= IBND(I)
      V  IEO=1
      V  IF(MOD(N13(I),2).EQ.0) IEO=2
      V3----DO 2500 J=1,3
      CC X(I,J)=TB(I,J)
      CC X(I,J)=TB(I,J)
      CC ACC(J)=0.0
      CC ACC(I,J)=0.0
      CC DXYZ=X(I,J)-TB(I,J)
      C.TN ENGL=ENGSP-A(L,J,IEO)*DXYZ-0.5DO*B(L,J,IEO)*DXYZ**2
      V3 WENGSP=WENGSP-A(L,J,IEO)*DXYZ-0.5DO*B(L,J,IEO)*DXYZ**2
      C.TN ENGL=ENGL-C(L,J,IEO)*(VX(I,J)-X(I,J))*2
      V3 WENG=WENG-C(L,J,IEO)*(VX(I,J)-X(I,J))*2
      V3 ACC(I,J)=A(L,J,IEO)+B(L,J,IEO)*DXYZ+C(L,J,IEO)*(VX(I,J)-X(I,J)),
      C2500 ACC(J)= ACC(J) + ACC(I,J)
      V3-2500 ACC(I,J)= ACC(I,J) + ACC(I,J)
      C
      V  IFLIFLG(7).NE.-1)
      V  IF(L1240.NE.-1)
      C
      3000 CONTINUE
      C
      V3----DO 3100 J=1,3
      C3100 FX1(J,I)=AC(I,J)
      C3100 FX1(J,I)=AC(I,J)
      C
      3200 CONTINUE
      C
      FIXING FORCE.
      C
      V3----DO 3250 J=1,3
      C3250 ACC(J)=AC(J)-FX1(J,I)
      C3250 ACC(I,J)=AC(I,J)-FX1(J,I)
      INNER PARTICLE.
      C
      3000 CONTINUE
      C
      3400 CONTINUE
      C
      *****PARAMETER OF MOTION *****
      C
      C
      C
      CXXXXXX
      CXXXXXXXXXXXX
      * WRITE(6,6200) 1, DDT, AC, (VX(I,J),J=1,3)
      * 6200 FORMAT(1H ,'*'GR240**', 2X, 'I=','I3', 'DDT=', 'F8.5,2X,'AC='
      * , 3(1PD15.7), 2X, 'VX=','3(1PD15.7)')

FORTRAN 77/VFP V12L10
-----7-----6-----5-----4-----3-----2-----1-----8
      00001570
      00001580
      00001590
      00001600
      00001610
      00001610
      00001620
      00001630
      00001640
      00001650
      00001660
      00001670
      00001670
      00001680
      00001690
      00001700
      00001710
      00001720
      00001730
      00001740
      00001750
      00001760
      00001761
      00001770
      00001780
      00001790
      00001800
      00001800
      00001810
      00001810
      00001820
      00001830
      00001840
      00001850
      00001860
      00001870
      00001880
      00001890
      00001900
      00001910
      00001920
      00001930
      00001940
      00001950
      00001960
      00001970
      00001980
      00001990
      00002000
      00002010
      00002020
      00002030
      00002040
      00002050

```

Fig.5.4 Modified Subroutine GR240 (Continued)

```

★ GR240  ( 5 )

      FILE=J9127.ZGR12DVP.FORT77(GR240)  FORTRAN 77/VP V12L10
      C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
      CXXXXXXXXXXXXXX
      1 2-----V3-----DO 3500 J=1,3
      1 2-----C.TN  VX(I,J)= 2.000*(ACC(J)*DDT+VX(I,J))- X(I,J)
      1 2-----V3  VX(I,J)= 2.000*(ACC(I,J)*DDT+VX(I,J))- X(I,J)
      1 +-----V3-3500 CONTINUE
      CXXXXXXXXXXXXXX
      1 *      WRITE(6,6300) X(35,1), (X(I,J),J=1,3), (VX(I,J),J=1,3)
      1 *6300 FORMAT(1H ',',*GR240*,1PD15.7,'X =, , 3(1PD15.7),*VX =,
      1 *          3(1PD15.7))
      CXXXXXXXXXXXXXX
      1
      1 +-----V--5000 CONTINUE
      C.TN  ENGSP = ENGSP + WENGSP
      SSN=0116  ENGL = ENGL + WENGL
      SSN=0117
      SSN=0118
      C
      1240=0
      C
      RETURN
      END
      SSN=0119
      SSN=0120
      SSN=0121
      SSN=0122
      SSN=0123
      SSN=0124

```

Fig. 5.4 Modified Subroutine GR240 (Continued)

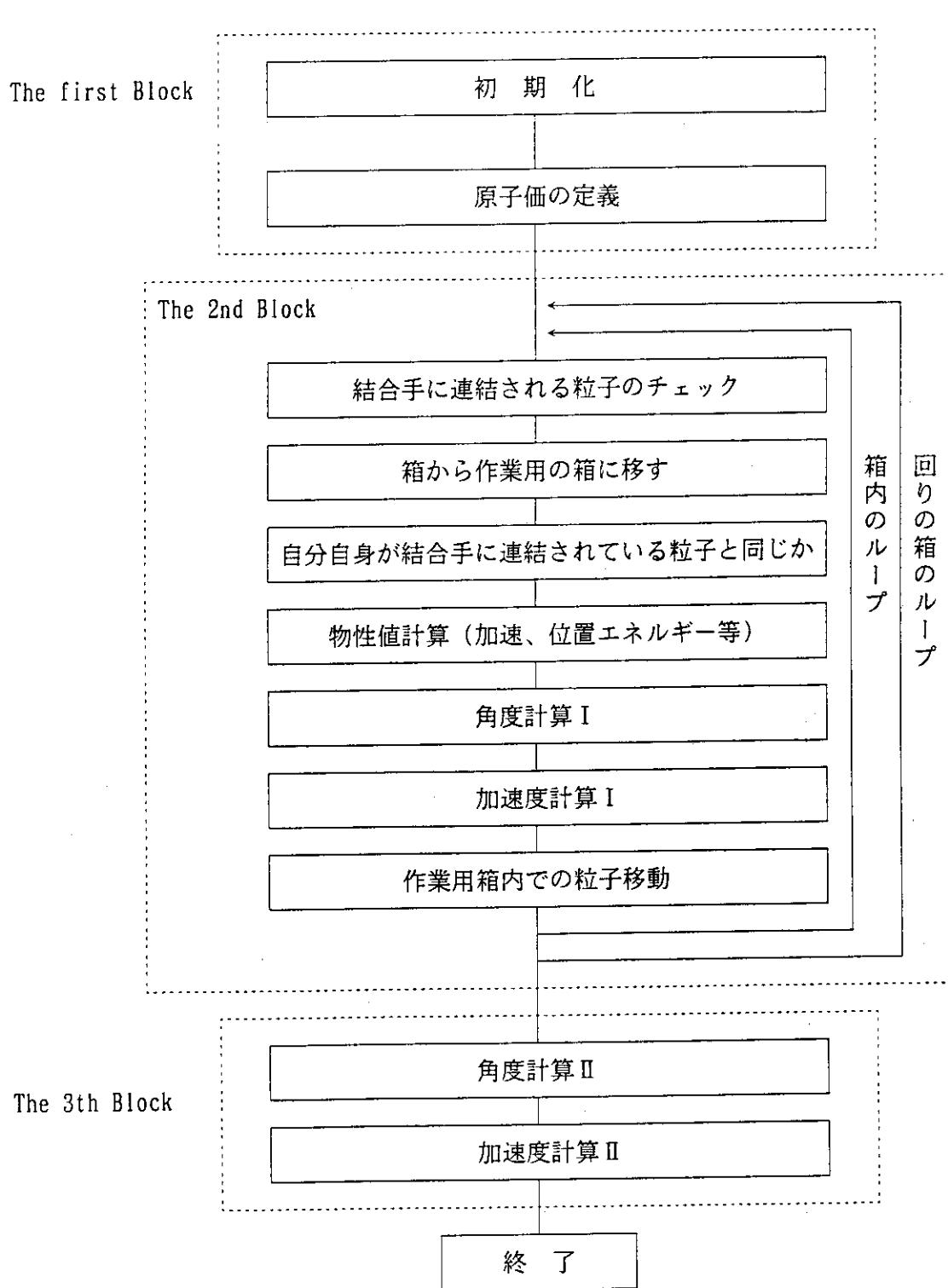


Fig.5.5 Computation Flow of Subroutine GR2D41

Fig. 5.6 Subroutine GR241

```

FILE=J9127.ZGR12D.FORT77(GR241)
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      DATA CP2 / 2.73081D0 /
C
C   EQUIVALENCE ( IIFLG( 4) ,  IDLT )
C   EQUIVALENCE ( CNTRL( 6) ,  N )
C   EQUIVALENCE ( DNYA( 1) ,  X(1,1) )
C   EQUIVALENCE ( DNYA( 48001) ,  AC(1) )
C   EQUIVALENCE ( DNYA( 48007) ,  Z21(1) )
C   EQUIVALENCE ( DNYA( 56007) ,  Z22(1) )
C   EQUIVALENCE ( DNYA( 64007) ,  Z23(1) )
C   EQUIVALENCE ( GEOM( 1) ,  IALP )
C   EQUIVALENCE ( GEOM( 2) ,  IBET )
C   EQUIVALENCE ( GEOM( 3) ,  IGAM )
C   EQUIVALENCE ( GEOM( 4) ,  TB(1,1) )
C   EQUIVALENCE ( GEOM( 12004) ,  IBND(1) )
C   EQUIVALENCE ( GEOM( 16004) ,  IK(1) )
C   EQUIVALENCE ( GEOM( 18739) ,  IAUX(1) )
C   EQUIVALENCE ( GEOM( 18766) ,  IDUX(1) )
C   EQUIVALENCE ( GEOM( 18793) ,  IBOX(1,1) )
C   EQUIVALENCE ( WORK( 1) ,  IBND(1) )
C   EQUIVALENCE ( WORK( 4001) ,  ICNV(1) )
C
C   RR(XW,YW,ZW) = DSQRT( XW**2+YW**2+ZW**2 )          * FUNCTION *
C
C   ----- INITIAL SET. -
C
C   IA = IALP-1
C   IB = IBET/IALP-1
C   IC = IGAM/IBET-1
C
C   DO 300 I=1,3
C     NF(I) = 0
C     RNFC(I) = 0.0D0
C   DO 100 J=1,3
C     R(J,I) = 0.0D0
C   100 CONTINUE
C   300 CONTINUE
C
C   AC(1) = Z21(N0)
C   AC(2) = Z22(N0)
C   AC(3) = Z23(N0)
C
C   IO = IBND(N0)
C   IF( IO.LT.0 ) IO = 0
C
C   NVAL = 3
C   IF(CIO.NE.-3) GO TO 400
C   ITB=INT(TB(N0,-3)/CP2+0.5D0)
C   NVAL=MOD(ITB,-2)+1
C   400 IF(CIO.NE.-5) GO TO 410
C   ITB=INT(TB(N0,-3)/CP2+0.5D0)
C   NVAL=MOD(ITB,-2)+1
C   410 IF(CIO.EQ.-7) NVAL=1
C   IF(CIO.NE.-9) GO TO 420
C   ITB=INT(TB(N0,-3)/CP2+0.5D0)
C   NVAL=MOD(ITB,-2)
C   420 IF(CIO.EQ.-10) NVAL=1
C   IF(CIO.EQ.-11) NVAL=1
C

```

Fig. 5.6 Subroutine GR241 (continued)

```

★ GR241 ( 3 )

FILE=J9127.ZGR12D.FORT77(GR241)
-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
  IF(C10.EQ.13) NVAL=1          00001140
  IF(C10.EQ.16) NVAL=1          00001150
  IF(C10.EQ.18) NVAL=1          00001160
  IF(C10.EQ.19) NVAL=1          00001170
  IF(C10.EQ.20) NVAL=1          00001180
  IF(C10.EQ.22) NVAL=1          00001190
  IF(C10.EQ.23) NVAL=1          00001200
  IF(C10.EQ.24) NVAL=1          00001210
  IF(C10.EQ.26) NVAL=1          00001220
  IF(C10.NE.8)                GO TO 430 00001230
  ITB=INT(TB(NO,3)/CP2+0.5D0),
  NVAL=1-MOD(ITB,2)            00001240
  430  IF(C10.EQ.21) NVAL=0      00001250
  IF(C10.EQ.25) NVAL=0      00001260
  IF(C10.NE.2)                GO TO 440 00001270
  ITB=INT(TB(NO,3)/CP2+0.5D0),
  NVAL=2-MOD(ITB,2)            00001280
  440  IF(C10.EQ.6)             GO TO 450 00001290
  ITB=INT(TB(NO,3)/CP2+0.5D0),
  NVAL=2-MOD(ITB,2)            00001300
  450  IF(C10.EQ.12) NVAL=2      00001310
  IF(C10.EQ.14) NVAL=2      00001320
  IF(C10.EQ.15) NVAL=2      00001330
  IF(C10.EQ.17) NVAL=2      00001340
  C   IF(NO.GT.N-NR) NVAL=0      00001350
  C   ICOMB=0                  GO TO 600 00001360
  IF(KK.EQ.0)                  GO TO 600 00001370
  NEWB=0                      00001380
  1----DO 460 J=1,KK           00001390
  1----IF(C111(J).EQ.NO) NEWB=NEWB+1 00001400
  1----460  IF(C114(J).EQ.NO) NEWB=NEWB+1 00001410
  NEWB=NEWB/2
  NVAL=NVAL-NEWB
  C   WRITE(6,6691) NO,NVAL,NR          DEBUG
  C6691  FORMAT('1H ',* DEBUG -- SUBROUTINE GR241,*',
  C   '1H ','NO',*15,
  C   '1H ','NVAL',*15,
  C   '1H ','N',*15,
  C   '1H ','NR',*15)
  C   600  J1 = LIBND(NO)
  LIBND(NO) = 0
  C   I2 = IAUX(C10+1)
  C   IC = IAUX(C10+1)+IDUX(C10+1)-1
  C   LL=1
  1----DO 750 I=1,500           * THE LAST ENTRY OF TABLE *
  1----750  IBOXNM(I)=0
  C
  C   WRITE(6,6601) J1,I2,IC          DEBUG
  C6601  FORMAT('1H ',* DEBUG -- SUBROUTINE GR241,*',
  C

```

Fig.5.6 Subroutine GR241 (continued)

```

★ GR241 < 4 >

FILE=J9127.ZGR12D.FORT77(GR241)
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C      /1H  'J1   '15,
C      /1H  'I2   '15,
C      /1H  'IC   '15,
C
C-----*----- DEBUG.
C-----*----- NEAR BOX. LOOP
C-----*----- NEAR BOX NUMBER *
C
800 CONTINUE
C      I3 = J1+IK(I2)          BOX OVER ?
C
C      IF( I3.LE.0 .OR. I3.GT.4096 )          DO 900 I=1,4
C      IBOXWI = IBOX(I,I3)          IBOXW(I,JJJ)=I4
C
C-----*----- DEBUG.
C      WRITE(6,6602) IBOXW(JJJ),JJJ=1,4
C6602 FORMAT(1H'*' DEBUG --- SUBROUTINE GR241 '*'
C      /1H  'IBOXW(1--4)  '415)          DEBUG.
C-----*----- LOOP BY 4-PARTICLE *
C
1000 CONTINUE
C      I4 = IBOXW(4)          GO TO 1800
C      IF( I4.EQ.0 )          GO TO 1700--OBJECT PARTICLE(I4) IS NO--SELF ?00001950
C
C      IF( NO.EQ.I4 )          GO TO 1700
C
C-----*----- DO 1010 I=1,LL
C      IBOXWW(LL)=I4
C
C      LL=LL+1
C
C      IF( IBOXWW(LL)=I4
C
C      XW = X(I4,1)-X(NO,1)
C      YW = X(I4,2)-X(NO,2)
C      ZW = X(I4,3)-X(NO,3)
C      WRK = RRC(XW,YW,ZW)
C
C      IF( WRK.GT.TOL )          FAR ? OR EVER BEEN CALCULATED?00002070
C      GO TO 1700          DPHAI4/DRO
C
C      IPASS=0
C      CALL GR247A(WRK,XW,YW,ZW,NO,I4,IPASS)
C
C      IF( NO.GT.I4 )          GO TO 1100
C
C      IF( WRK.LT.TIK )          GO TO 1100--METALLIC INTERACTION
C      CALL GR247(WRK,XW,YW,ZW,I4)
C
C
1100 CONTINUE
C      IF( IPASS.EQ.1 )          GO TO 1700
C      IF(ICONV(I4).GE.3)      GO TO 1700
C      IF(ICONV(NO).GE.3)      GO TO 1700
C      IF(I4.GT.N-NR)          GO TO 1700
C
C      NVAL1 = ICOMB+1
C      DO 1200 L=NVAL1,NVAL
C      IF( RNF(CL).EQ.0.0DO .OR. RNF(CL).GT.WRK )
C
1

```

Fig.5.6 Subroutine GR241 (continued)

```

FILE=J9127.ZGR12D.FORT77(GR241)
1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      *-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
      -1200 CONTINUE
      GO TO 1300
      C 1300 CONTINUE
      C
      IF( I4 .EQ. NF(1) ) GO TO 1700
      J = NVAL
      C 1400 CONTINUE
      IF( J .EQ. L )
      NF(J) = NF(J-1)
      RNFC(J) = RNFC(J-1)
      DO 1500 I=1,3
      R(I,J) = R(I,J-1)
      CONTINUE
      J = J-1
      GO TO 1400 * SET OF NEAREST PARTICLE DATA *
      C 1600 CONTINUE
      NF(L) = I4
      RNFL(L) = WRK
      R(1,L) = XW
      R(2,L) = YW
      R(3,L) = ZW
      C
      1700 CONTINUE
      IBOXW(4) = IBOXW(3)
      IBOXW(3) = IBOXW(2)
      IBOXW(2) = IBOXW(1)
      IBOXW(1) = 0
      C
      1800 CONTINUE
      I2 = I2+1
      C 1F( I2.LE.IC ) GO TO 800 * CHECK OF NEARBY BOX NUMBER *
      C
      DO 1900 I=1,3
      IF(NF(I).NE.0) GO TO 1850
      1850 CONTINUE
      -1900 CONTINUE
      IF( NO.GT.N-NR) GO TO 2300
      C
      WRITE(6,6604) (IBOXW(I),I=1,500)
      C 6604 FORMAT(1H '* DEBUG -- SUBROUTINE GR241 *',
      C           ' /1H '*IBOXW(1--500)'
      C           ' 25C(1H 2015)')
      C
      WRITE(6,6699) NO, (RNFC(I),I=1,3), ((R(I,J),I=1,3),J=1,3),
      CNF(1),I=1,3)
      C 6699 FORMAT(1H '* DEBUG -- SUBROUTINE GR241 *',
      C           ' /1H '*NO
      C           ' ,IS,
      C
      C 10 1700 REGARD I4 AS NEIGHBOUR
      C
      ALREADY REGISTERED
      AS NEW COMBINE ?
      C
      00002370
      00002380
      00002390
      00002400
      00002410
      00002420
      00002430
      00002440
      00002450
      00002460
      00002470
      00002480
      00002490
      00002500
      00002510
      00002520
      00002530
      00002540
      00002550
      00002560
      00002570
      00002580
      00002590
      00002600
      00002610
      00002620
      00002630
      00002640
      00002650
      00002660
      00002670
      00002680
      00002690
      00002700
      00002710
      00002720
      00002730
      00002740
      00002750
      00002760
      00002770
      00002780
      00002790
      00002800
      00002810
      00002820
      00002830
      00002840
      00002850
      00002860
      00002870
      00002880
      00002890
      00002900
      00002910
      00002920
      00002930

```

Fig. 5.6 Subroutine GR241 (continued)

```

★ GR241  < 6 >

C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      FILE=19127,ZGR12D,FORT77(GR241)
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      /1H , 'RNF(1-3)=' , 3(1X,D23-16) ,
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      /1H , 'RC1-3/1)' , 3(1X,D23-16) ,
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      /1H , 'RC1-3/2)' , 3(1X,D23-16) ,
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      /1H , 'RC1-3/3)' , 3(1X,D23-16) ,
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      /1H , 'NFC(1-3)' , X315 ) ----- DEBUG .
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      ANALYTICAL METHOD
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      CALL GR245( 1240, NO, RNF, R, NVAL )
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      2300 CONTINUE
      RETURN
      END

```

Fig.5.6 Subroutine GR241 (continued)

```

#####
## GR241   ##
##### SSN=0001 #####
##### SSN=0002 #####
FILE=J9127.ZGR12DVP.FORT77(GR241)          FORTRAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      SUBROUTINE GR241
      SUBROUTINE GR241 ( 1240, NO )
      IMPLICIT REAL*8 ( A-H,O-Z )
C-----*
C 1. FUNCTION.
C 2. INPUT.
      1240    INITIAL SWITCH OF *FCC240*.
      NO       CENTER PARTICLE NUMBER.
      C      X-COMPONENT OF ACCELERATION SUMED UP.
      C      Y-COMPONENT OF ACCELERATION SUMED UP.
      C      Z-COMPONENT OF ACCELERATION SUMED UP.
      C      IBND(4000) BOUNDARY TYPE OF PARTICLES.
      C      MODIFY IBND(--) TO INCLUDE NOT ONLY BOUNDARY TYPE.
      C      IBND(1075) GENERAL NEIGHBORS IN A 5X5X5 BOX ABOUT A GENERAL.
      C      IAUX(27) TABLE OF THE ENTRY TO EACH TABLE IN IK(--).
      C      IDUX(27) TABLE OF THE LENGTH OF EACH TABLE.
      C      IBOX(<-->) BOXES WHICH CONTAIN PARTICLE NUMBER.
      C      X(40000,3) COORDINATES OF PARTICLES.
      C      IALP (IA+1) IA;THE LENGTH OF THE CRISTAL ALONG X-CO.
      C      IBET (IA+1)*(IB+1)
      C      IGAM (IA+1)*(IB+1)*(IC+1)
      C      TB(4000,3) COORDINATS OF BOUNDARY PARTICLES.
C 3. OUTPUT.
      C      NOTHING.
C 4. LOCAL VARIABLES.
      C      NF(12) PARTICLE NUMBER OF TWELVE NEIGHBOURS.
      C      RNFX(12) DISTANCE BETWEEN THE INDICATED PARTICLE AND EACH
      C      NEIGHBOUR PARTICLE.
      C      R(3,12) RELATIVE COORDINATE OF 12 NEIGHBOUR PARTICLES.
      C      IGRP(4,12) PARTICLE NO. OF 12 NEIGHBOUR PARTICLES.
      C      THETA(4,12) ANGLE BETWEEN PARTICLE.
      C      WRNF(12) INTERVAL BETWEEN PARTICLE.
C 5. CALLED BY GR240.
C 6. CALLS GR245,GR247,GR247A
C-----*
C.TN
      DATA 1240/0/
      COMMON / VEC1 / AC(4000,3),1DC
      DIMENSION IC(4000),I2(4000),NVAL(4000),ICOMB(4000),J1(4000),
      * COMMON VARIABLES *          00000370
      C      DIMENSION Z72(6000),Z73(6000),X(4000,3),TB(4000,3),
      *           IBND(4000),IIBND(4000),IBOX(4,-098),ICONV(4000), 00000380
      *           IK(2735),IAUX(27),IDUX(27)          00000390
      C.TN :           IK(2735),IAUX(27),IDUX(27),AC(3),          00000400
      C      DATA 1240/0/
      C.TN      COMMON / VEC1 / AC(4000,3),RNFX(3),IBOXW(4),IBOXW(500),
      * LOCAL VARIABLES *          00000420
      C.TN      DIMENSION NF(4000,3),RNFX(4000,3),R(4000,3,3), 00000430
      *           IBOXW(6),IBOXW(500)          00000430
      C      DIMENSION WRNF(3),RNFX(3),IBOXW(4),IBOXW(500),
      *           COMMON VARIABLES *          00000440
      C      REAL*8 X, Z1, Z22, Z23, RNFX, R, WRNF, XW, YW, ZW, RR, WRK,
      *           DX, DY, DZ, TOL, AC,          00000450
      C      COMMON / GEOM / GEOM (39176),
      *           SSN=0003          00000460
      C      SSN=0004          00000470
      C      SSN=0005          00000480
      C      SSN=0006          00000490
      C      SSN=0007          00000490
      C      SSN=0008          00000490
      C      SSN=0009          00000490

```

Fig.5.7 Modified Subroutine GR241

```

★ GR241 〈 2 〉

FILE=J9127.ZGR12DVP.FORT77(GR241)  FORTRAN 77/VP V12L10
C-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      / DYN A / DYN A (78978)
      / WORK / WORK (8000)
      / IIFLG / IIFLG(10)
      / COMBIN / KK,ILL1(50),ILL4(50),NR
      / CNTRL / CNTRL (30)

C     DATA TOI    / 4.000000 /
C     DATA TIK    / 2.400000 /
C     DATA CP2    / 2.73081D0/
C
C     EQUIVALENCE C IIFLG( 4 ) / IDLT   )
C     EQUIVALENCE C CNTRL( 6 ) / N      )
C     EQUIVALENCE C DYN A ( 1 ) / X      (1,1)  )
C     EQUIVALENCE C DYN A (48007) / Z21   (1)   )
C     EQUIVALENCE C DYN A (56007) / Z22   (1)   )
C     EQUIVALENCE C DYN A (64007) / Z23   (1)   )
C     EQUIVALENCE C DYN A (48001) / AC    (1)   )
C     EQUIVALENCE C GEOM ( 1 ) / IALP  )
C     EQUIVALENCE C GEOM ( 2 ) / IBET  )
C     EQUIVALENCE C GEOM ( 3 ) / IGAM  )
C     EQUIVALENCE C GEOM ( 4 ) / TB    (1,1)  )
C     EQUIVALENCE C GEOM (12004) / IBND (1)   )
C     EQUIVALENCE C GEOM (16004) / IK    (1)   )
C     EQUIVALENCE C GEOM (18739) / IAUX (1)   )
C     EQUIVALENCE C GEOM (18766) / IDUX (1)   )
C     EQUIVALENCE C GEOM (18793) / IBOX (1,1)  )
C     EQUIVALENCE C WORK ( 1 ) / IBND(1)   )
C     EQUIVALENCE C WORK( 4001) / ICNV(1)   )
C
C.TN   COMMON / AREA / AREA( 14 )
C.TN   * / POTE / POT( 4000 )
C.TN   * FOR VECTORIZATION, CHANGED
C.TN   * FROM PHAI11-PHAI13, PHAI12-PHAI23,DPHAI11-DPHAI13,DPHAI121-DPHAI123
C.TN   * DIMENSION E1(3),E2(3),EE(1000,3),E(1000,3),
C.TN   *          WR(3,3),RI(3),RJ(3),FM(3),
C.TN   *          THRO(3,3),AC1(3),AC2(3),
C.TN   *          E1(3),E2(3),AC(3),WR(3,3),RI(3),FM(3),
C.TN   *          PHAI1(3),PHAI2(3),PHAI1(3),PHAI2(3),AC1(3),AC2(3),
C.TN   *          PHAI1(3),PHAI2(3),PHAI1(3),PHAI2(3),AC1(3),AC2(3),
C.TN   *          REAL*8 RI,RJ,THRO,PHAI,ENGPO,
C.TN   *          EQUIVALENCE C DYN A (72007) / EE (1,1)  )
C.TN   *          C DYN A (75007) / E (1,1)  )
C.TN   *          C DYN A (78007) / E1 (1)   )
C.TN   *          C DYN A (78010) / E2 (1)   )
C.TN   *          C DYN A (78955) / ENGPO  )
C.TN   *          EQUIVALENCE C GEOM ( 4 ) / TB   (1,1)  )
C.TN   *          EQUIVALENCE C AREA ( 1 ) / N    )
C.TN   *          EQUIVALENCE C AREA ( 1 ) / NS   )
C.TN   *          C AREA ( 2 ) / NSTP  )
C.TN   *          C AREA ( 3 ) / X1   )
C.TN   *          C AREA ( 5 ) / Y1   )
C.TN   *          C AREA ( 7 ) / Z1   )
C.TN   *          C AREA ( 9 ) / X2   )

SSN=0011
SSN=0012
SSN=0013

SSN=0014
SSN=0015
SSN=0016

SSN=0017

SSN=0018

SSN=0019

SSN=0020

SSN=0021
SSN=0022
SSN=0023

SSN=0024
SSN=0025

```

Fig.5.7 Modified Subroutine GR241 (continued)

Fig. 5.7 Modified Subroutine GR241 (continued)

```

★ GR241 ( 4 )

FILE=J9127-ZGR12DVP.FORT77(GR241)  FORT77/VP V12L10
C-----1-----2-----3-----4-----5-----6-----7-----8-----*
V 400 IF(CIO.NE.-5)          GO TO 410      00001050
V ITB=INT(TB(NO/3)/CP2+0.5D0)
V C.TN  NVAL=MOD(ITB/2)+1    00001060
V NVAL(NO)=MOD(ITB/2)+1    00001070
V C.410 IF(CIO.EQ.-7) NVAL=1    00001080
V 410 IF(CIO.EQ.-7) NVAL(NO)=1  00001080
V ITB=INT(TB(NO/3)/CP2+0.5D0) 00001090
V C.TN  NVAL=MOD(ITB/2)      00001090
V NVAL(NO)=MOD(ITB/2)      00001100
V C.420 IF(CIO.EQ.-10) NVAL=1   00001100
V 420 IF(CIO.EQ.-10) NVAL(NO)=1 00001120
V C.TN  IF(CIO.EQ.-11) NVAL=1   00001130
V IF(CIO.EQ.-11) NVAL(NO)=1  00001130
V C.TN  IF(CIO.EQ.-13) NVAL=1   00001140
V IF(CIO.EQ.-13) NVAL(NO)=1  00001140
V C.TN  IF(CIO.EQ.-16) NVAL=1   00001150
V IF(CIO.EQ.-16) NVAL(NO)=1  00001150
V C.TN  IF(CIO.EQ.-18) NVAL=1   00001160
V IF(CIO.EQ.-18) NVAL(NO)=1  00001160
V C.TN  IF(CIO.EQ.-19) NVAL=1   00001170
V IF(CIO.EQ.-19) NVAL(NO)=1  00001170
V C.TN  IF(CIO.EQ.-20) NVAL=1   00001180
V IF(CIO.EQ.-20) NVAL(NO)=1  00001180
V C.TN  IF(CIO.EQ.-22) NVAL=1   00001190
V IF(CIO.EQ.-22) NVAL(NO)=1  00001190
V C.TN  IF(CIO.EQ.-23) NVAL=1   00001200
V IF(CIO.EQ.-23) NVAL(NO)=1  00001200
V C.TN  IF(CIO.EQ.-24) NVAL=1   00001210
V IF(CIO.EQ.-24) NVAL(NO)=1  00001210
V C.TN  IF(CIO.EQ.-26) NVAL=1   00001220
V IF(CIO.EQ.-26) NVAL(NO)=1  00001220
V C.TN  IF(CIO.EQ.-8) NVAL=1   00001230
V IF(CIO.EQ.-8) NVAL(NO)=1  00001240
V ITB=INT(TB(NO/3)/CP2+0.5D0)
V C.TN  NVAL=1-MOD(ITB/2)    00001250
V NVAL(NO)=1-MOD(ITB/2)    00001260
V C.430 IF(CIO.EQ.-21) NVAL=0  00001270
V 430 IF(CIO.EQ.-21) NVAL(NO)=0 00001270
V C.TN  IF(CIO.EQ.-25) NVAL=0  00001270
V IF(CIO.EQ.-25) NVAL(NO)=0  00001270
V C.TN  IF(CIO.EQ.-2) NVAL=0   00001280
V ITB=INT(TB(NO/3)/CP2+0.5D0)
V C.TN  NVAL=2-MOD(ITB/2)    00001290
V NVAL(NO)=2-MOD(ITB/2)    00001300
V C.TN  IF(CIO.EQ.-6) NVAL=0   00001310
V IF(CIO.EQ.-6) NVAL(NO)=0  00001320
V C.TN  NVAL=2-MOD(ITB/2)    00001330
V NVAL(NO)=2-MOD(ITB/2)    00001340
V C.450 IF(CIO.EQ.-12) NVAL=2  00001350
V 450 IF(CIO.EQ.-12) NVAL(NO)=2 00001350
V C.TN  IF(CIO.EQ.-14) NVAL=2  00001360
V IF(CIO.EQ.-14) NVAL(NO)=2  00001360
V C.TN  IF(CIO.EQ.-15) NVAL=2  00001370
V IF(CIO.EQ.-15) NVAL(NO)=2  00001370
V C.TN  IF(CIO.EQ.-17) NVAL=2  00001380
V IF(CIO.EQ.-17) NVAL(NO)=2  00001380

```

Fig.5.7 Modified Subroutine GR241 (continued)

```

★ GR241 ( 5 )

FILE=J9127-ZGR12DVP.FORT77(GR241)          FORTRAN 77/VP V12L10
C---*---1 IF( NO .GT. N-NR ) NVAL=0           *---2---*---3---*---4---*---5---*---6---*---7---*---8
C.TN  IF( NO .GT. N-NR ) NVAL=0           00001390
V   C. TN  ICOMB( NO )=0           00001400
C. TN  ICOMB( NO )=0           00001410
V   C. TN  NEWB=0           00001420
C. TN  NEWB=0           00001430
C.TN  DO 460 J=1,KK           00001440
C.TN  IF( C1111( J ) .EQ. NO ) NEWB=NEWB+1           00001450
C. TN  IF( C1114( J ) .EQ. NO ) NEWB=NEWB+1           00001460
C.TN  NEWB=NEWB/2           00001470
C.TN  NVAL( NO )=NVAL( NO )-NEWB           00001480
C---*--- WRITE( 6,6691 ) NO ,NVAL( NO ),NR           DEBUG.
C6691  FORMAT( 1H ,* DEBUG -- SUBROUTINE GR241 *,'
C   1H , 'NO '
C   1H , 'NEWB '
C   1H , 'NVAL '
C   1H , 'N '
C   1H , 'NR '
C---*--- C. 600 J1 = LIBND( NO )           DEBUG.
V   600 J1( ND ) = LIBND( NO )
V   C. TN  I2 = IAUX( IO+1 )
V   C. TN  I2( NO ) = IAUX( IO+1 )
V   C. TN  IC = IAUX( IO+1 )+IDUX( IO+1 )-1           * THE LAST ENTRY OF TABLE *
V   C. TN  IC( NO ) = IAUX( IO+1 )+IDUX( IO+1 )-1
V   C. TN  CONTINUE
C---*--- SSN=0095
C---*--- SSN=0096
C---*--- SSN=0097
C---*--- SSN=0098
C---*--- SSN=0099
C---*--- SSN=0100
C---*--- SSN=0101
C---*--- SSN=0102
C---*--- SSN=0103
C---*--- SSN=0104
C---*--- SSN=0105
C---*--- SSN=0106
C---*--- SSN=0107
C---*--- SSN=0108
C---*--- SSN=0109
C---*--- SSN=0110
C---*--- SSN=0111
C---*--- SSN=0112
C---*--- SSN=0113
C---*--- SSN=0114
C---*--- SSN=0115
C---*--- SSN=0116
C---*--- DO 4300 NO=1, N
C---*--- AC( NO,1 ) = Z21( NO )
C---*--- AC( NO,2 ) = Z22( NO )
C---*--- AC( NO,3 ) = Z23( NO )
C---*--- LL=1
C---*--- DO 750 I=1,500
C---*--- S---750 IBOXWM( 1 )=0
C---*--- C. WRITE( 6,6601 ) J1( NO ),I2( NO ),IC( NO )           DEBUG.
C6601  FORMAT( 1H ,* DEBUG -- SUBROUTINE GR241 *,'
C   1H , J1 , ,IS

```

Fig.5.7 Modified Subroutine GR241 (continued)

JAERI-Data/Code 94-021

```

FILE=J9127.ZGR12DVP.FORT77(GR241)          FORTAN 77/VP V12L10
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C      .      /1H   '12
C      .      /1H   '1C
C      .      '15
C      .      '15
C-----*----- DEBUG
C-----*----- NEAR BOX. LOOP
C-----*----- * NEAR BOX NUMBER *
C
C     800 CONTINUE
C     TN 13 = J1+IK(I2)
C     TN 13 = J1(NO)+IK(I2(NO))
C
C     IF( 13.LE.0 .OR. 13.GT.4096 )           BOX OVER ?
C
C     C
C     2-----DO 900 I=1,4
C     2-----IBOXW(C1) = IBOX(I1,I3)
C
C     4-----DO 900 CONTINUE
C     4-----IBOXW(C1) = IBOX(I1,I3)
C----- DEBUG.
C     WRITE(6,6602) CIBOXW(CJJJ),JJJ=1,4
C     6602 FORMAT(1H '/* DEBUG */',I1H '/* SUBROUTINE GR241 */',
C               '1H '/* IBOXW(I1--4) */',4,15)
C----- DEBUG.
C----- * LOOP BY 4-PARTICLE *
C
C     1000 CONTINUE
C     14 = IBOXW(4)
C     IF( 14.EQ.0 )           GO TO 1800
C----- OBJECT PARTICLE(I4) IS NO-SELF ? 000001950
C     IF( NO.EQ.I4 )           GO TO 1700
C
C     2-----DO 1010 I=1,LL
C     2-----1010 IF( IBOXWW(I1).EQ.I4 )
C               LL=LL-1
C               IBOXWW(LL)=I4
C
C     C DEBUG
C
C     N=0123
C     N=0124
C     N=0125
C     N=0126
C
C     N=0127
C     N=0128
C     N=0129
C     N=0130
C
C     N=0131
C     N=0132
C     N=0133
C     N=0134
C
C     N=0135
C     N=0136
C
C     N=0137
C     N=0138
C     N=0139
C
C     N=0140
C
C     N=0141
C
C     N=0142
C     N=0143
C
C     N=0144
C
C     N=0145
C
C     N=0146
C
C     N=0147
C
C     N=0148
C
C     N=0149
C
C     N=0150
C
C     C IPASS=1
C     C CALL GR247A(WRK,XW,YW,ZW,NO,I4,IPASS)
C
C     C IF( WRK.GT.101 )           GO TO 1700
C----- DPHAI4/DRO
C
C     C IPASS=0
C     TN CALL GR247A(WRK,XW,YW,ZW,NO,I4,IPASS)
C
C     C IF( KK.EQ.0 )           GO TO 4000
C----- DO 4000 I=1,KK
C     C IF( III1(I1).EQ.NO .AND.
C           III4(I1).EQ.I4 )           GO TO 2500
C     C IF( III1(I1).EQ.I4 .AND.
C           III4(I1).EQ.NO )           GO TO 2500
C
C     C GO TO 4000
C
C     2500 IPASS=1
C     IF( NO.LT.I4 )           GO TO 3000
C               GO TO 4000
C----- DPHAI4/DRO
C
C     C3000 CONTINUE
C     CC C1=EXP(-2.0D0*ALP3*(WRK-RO3))

```

Fig. 5.7 Modified Subroutine GR241 (continued)

```

★ GR241 ( 7 )

FILE=J9127-ZGR12DVP.FORT77(GR241) FORTRAN 77/VP V12L10
C---*---1-----*---2-----*---3-----*---4-----*---5-----*---6-----*---7-----*---8
CC C2=2.0D0*DEXP(-ALP3*(WRK-RO3))
CC F=D3*(C1-C2)
CC DF=D3*ALP3*(2.0D0*C1-C2)
3000 F = -D12 + D13*(WRK - RO3)**2
DF = 2.0D0*D13*(WRK - RO3)
C----- AC(1)=AC(1)+DF*(XW/WRK)
C.TN AC(2)=AC(2)+DF*(YW/WRK)
C.TN AC(3)=AC(3)+DF*(ZW/WRK)
C.TN AC(NO,1)=AC(NO,1)+DF*(XW/WRK)
AC(NO,2)=AC(NO,2)+DF*(YW/WRK)
AC(NO,3)=AC(NO,3)+DF*(ZW/WRK)
221(I4)=Z21(I4)-DF*(XW/WRK)
222(I4)=Z22(I4)-DF*(YW/WRK)
Z23(I4)=Z23(I4)-DF*(ZW/WRK)
C----- ENGP0=ENGP0+F ----- POTENTIAL ENERGY.
C----- WRITE(6,6000) NO,I4,I,II1(I),II4(I),WRK,F,DF,D12,D13,RO3
C6000 FORMAT(1H ,*, DEBUG --- SUBROUTINE GR247A *,*
C      ,'/1H ',NO
C      ,'/1H ',I4
C----- CONTINUE
4000 CONTINUE
4001 CONTINUE
C----- IF( NO .GT. I4 ) GO TO 1100
C----- IFC WRK.LT.TIK , GO TO 1100
C----- METALLIC INTERACTION
C.TN CALL GR247(WRK,XW,YW,ZW,I4)
C.TN CALL GR247(WRK,XW,YW,ZW,I4,NO)
C.TN IF(WRK.EQ.0.0D0) GO TO 1247
E11 = E1(3)*WRK+E2(3)
I5 = E11
IF( I5 .LT. 1 ) I5 = 1
IF( I5 .GE. 1000 ) I5 = 999
H = E11-I5
C----- ACC = E(I5,3)+H*(E(I5+1,3)-E(I5,3))
C----- ACCELERATION.
C.TN AC(1) = AC(1)+ACC*XW/WRK
C.TN AC(2) = AC(2)+ACC*YW/WRK
C.TN AC(3) = AC(3)+ACC*ZW/WRK
C.TN AC(NO,1) = AC(NO,1)+ACC*XW/WRK
AC(NO,2) = AC(NO,2)+ACC*YW/WRK
AC(NO,3) = AC(NO,3)+ACC*ZW/WRK
C----- Z21(I4) = Z21(I4)-ACC*XW/WRK
Z22(I4) = Z22(I4)-ACC*YW/WRK
Z23(I4) = Z23(I4)-ACC*ZW/WRK
C----- ENGP0=ENGP0+EE(I5,3)+H*(EE(I5+1,3)-EE(I5,3))
1247 CONTINUE

```

Fig.5.7 Modified Subroutine GR241 (continued)

```

★ GR241 ( 8 )

FILE=J9127.2GR12DVP.FORT77(GR241) FORTRAN 77/VFP V12L10
C-----1-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8
C
1100 CONTINUE
  IF(CIPASS.EQ.1) GO TO 1700
  IF(ICONV(I4).GE.3) GO TO 1700
  IF(ICONV(I4).GE.3) GO TO 1700
  IF(ICONV(I4).GT.N-NR) GO TO 1700
C
C     NEAREST PARTICLE NUMBER.
C   TN  NVAL1 = ICOMB(NO)+1
NVAL1 = ICOMB(NO)+1
C   TN  DO 1200 L=NVAL1,NVAL
DO 1200 L=NVAL1,NVAL(NO)
C   TN  IFC(RNF(L).EQ.0.0D0.OR.RNF(L).GT.WRK)
IFC(RNF(NO,L).EQ.0.0D0.OR.RNF(NO,L).GT.WRK)
GO TO 1300
C
1200 CONTINUE
GO TO 1700  REGARD I4 AS NEIGHBOUR
C
C   1300 CONTINUE
C
C   TN  IFC I4 .EQ. NF(1) ) GO TO 1700
C   TN  IFC I4 .EQ. NF(NO,1) ) GO TO 1700
C   TN  J = NVAL(NO)
C
1400 CONTINUE
IFC(J.EQ.L) GO TO 1600
C   TN  NF(J) = NF(J-1)
NF(NO,J) = NF(NO,J-1)
C   TN  RNF(CL) = RNF(CL-1)
RNF(NO,J) = RNF(NO,J-1),
DO 1500 I=1,3
C   TN  R(I,J) = R(I,J-1)
R(NO,I,J) = R(NO,I,J-1),
C
1500 CONTINUE
J = J-1
GO TO 1400 * SET OF NEAREST PARTICLE DATA *
C
1600 CONTINUE
C   TN  NF(CL) = I4
NF(NO,L) = I4
C   TN  RNF(CL) = WRK
RNF(NO,L) = WRK
C   TN  R(1,L) = XW
R(NO,1,L) = XW
C   TN  R(2,L) = YW
R(NO,2,L) = YW
C   TN  R(3,L) = ZW
R(NO,3,L) = ZW
C
1700 CONTINUE
IBOXW(4) = IBOXW(3)
IBOXW(3) = IBOXW(2)
IBOXW(2) = IBOXW(1)
IBOXW(1) = 0
C
SSN=0173
SSN=0174
SSN=0175
SSN=0176
SSN=0177
SSN=0178
SSN=0179
SSN=0180
SSN=0181
SSN=0182
SSN=0183
SSN=0184
SSN=0185
SSN=0186
SSN=0187
SSN=0188
SSN=0189
SSN=0190
SSN=0191
SSN=0192
SSN=0193
SSN=0194
SSN=0195
SSN=0196
SSN=0197
SSN=0198
SSN=0199
SSN=0200
SSN=0201
SSN=0202
SSN=0203
SSN=0204
SSN=0205

```

Fig.5.7 Modified Subroutine GR241 (continued)

Fig.5.7 Modified Subroutine GR241 (continued)

Figure 5.7 Modified Subroutine GR241 (continued)

```

★ GR241 ( 11 )

      FILE=J9127.ZGR12DVP.FORT77(GR241)
      C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      V 1250 CONTINUE
      C-TN CALL GR244(CRNF,THETA,PHAI)
      C
      V PHAI=0.0DO
      C-TN PHAI1(C1)=0.0DO
      C4000 DO 4000 I=1,3
      V PHAI1(C1)=0.0DO
      V PHAI11=0.0DO
      V PHAI12=0.0DO
      V PHAI122=0.0DO
      V PHAI123=0.0DO
      C-TN DO 2000 I=1,3
      C-TN RR=RNFC1
      CCCC1 IF(CRNF(C1).EQ.0.0) RR=RO
      C-TN IF(CRNF(C1).EQ.0.0DO) GO TO 2000
      C-TN E11=E1(C1)*RR+E2(C1)
      C-TN I5=E11
      C-TN IFC15.LT.-1 15=1
      C-TN IFC15.GE.-1000) 15=999
      C-TN H=E11-I5
      C-TN PHAI1(C1)=EE(15,1)+H*(EE(15+1,1)-EE(15,1))
      C-TN CONTINUE
      C2000 CONTINUE
      V RR4=RNFC(NO,1)
      V IF(CRNF(NO,1).NE.0.0DO) THEN
      V E11=E1(C1)*RR4+E2(C1)
      V I5=E11
      V IFC15.LT.-1 15=1
      V IFC15.GE.-1000) 15=999
      V H=E11-I5
      V PHAI11=EE(C15,1)+H*(EE(C15+1,1)-EE(C15,1))
      V
      V END IF
      V RR4=RNFC(NO,2)
      V IF(CRNF(NO,2).NE.0.0DO) THEN
      V E11=E1(C1)*RR4+E2(C1)
      V I5=E11
      V IFC15.LT.-1 15=1
      V IFC15.GE.-1000) 15=999
      V H=E11-I5
      V PHAI12=EE(15,1)+H*(EE(15+1,1)-EE(15,1))
      V
      V END IF
      V RR4=RNFC(NO,3)
      V IF(CRNF(NO,3).NE.0.0DO) THEN
      V E11=E1(C1)*RR4+E2(C1)
      V I5=E11
      V IFC15.LT.-1 15=1
      V IFC15.GE.-1000) 15=999
      V H=E11-I5
      V PHAI13=EE(15,1)+H*(EE(15+1,1)-EE(15,1))
      V
      V END IF
      C-TN DO 4400 I=1,3
      C-TN ANGLE=THETAC1
      C-TN IF(THETAC1.EQ.0.0DO) ANGLE=THETAO
      CCCC1 IF(THETAC1.EQ.0.0) GO TO 4000
      C-TN
      C-----*-----1-----*-----2-----*-----3-----*-----4-----*-----5-----*-----6-----*-----7-----*-----8-----*
      00001110
      00001120
      00000270
      00000280
      00000290
      00000300
      00000310
      00000300
      00000310
      00000300
      00000310
      00000300
      00000310
      00000300
      00000320
      00000310
      00000330
      00000340
      00000350
      00000360
      00000370
      00000380
      00000390
      00000400
      00000410
      00000420
      00000430
      00000340
      00000360
      00000370
      00000380
      00000390
      00000400
      00000410
      00000420
      00000340
      00000360
      00000370
      00000380
      00000390
      00000400
      00000410
      00000420
      00000440
      00000450
      00000460
      00000470
      00000480

```

Fig.5.7 Modified Subroutine GR241 (continued)

Fig.5.7 Modified Subroutine GR241 (continued)

Fig. 5.7 Modified Subroutine GR241 (continued)

Fig. 5.7 Modified Subroutine GR241 (continued)

Fig. 5.7 Modified Subroutine GR241 (continued)

```

★ GR241 ( 16 )

FILE=J9127_ZGR12DVP.FORT77(GR241) FORTRAN 77/VP V12L10
      *      *      *      *      *      *      *      *      *      *
      1      2      3      4      5      6      7      *      *      *
C5800 AC1(I)=-(DPHAI1*(1)*WR(1,1))*PHAI12*(3)          00001790
      C.TN   -PHAI1*(1)*DPHAI12*(3)*THR0(1,1)          00001800
      C.TN   -PHAI1*(1)*PHAI12*(1)*DPHAI12*(3)*THR0(1,3) 00001810
      C.TN   -AC11=-(DPHAI11*WR11)*PHAI121*PHAI123          00001790
      V      -PHAI11*DPHAI121*PHAI123*THR11          00001800
      V      -PHAI11*PHAI121*DPHAI123*THR13          00001810
      V      AC12=-(DPHAI11*WR21)*PHAI121*PHAI123          00001790
      V      -PHAI11*DPHAI121*PHAI123*THR21          00001800
      V      AC13=-(DPHAI11*WR31)*PHAI121*PHAI123          00001790
      V      -PHAI11*DPHAI121*PHAI123*THR31          00001800
      C-----*
      C      DO 1820 I=1,3
      C      C1820 AC2(I)=-(DPHAI1*(2)*WR(1,2))*PHAI12*(2)*PHAI12*(1)
      C      -PHAI1*(2)*DPHAI12*(2)*PHAI12*(1)*THR0(1,2) 00001830
      C.TN   C.TN   -PHAI1*(2)*DPHAI12*(1)*THR0(1,1) 00001840
      C.TN   C.TN   -AC21=-(DPHAI12*WR12)*PHAI122*PHAI121 00001850
      V      V      -PHAI12*DPHAI122*PHAI121*THR11 00001860
      V      V      -AC22=-(DPHAI12*WR22)*PHAI122*PHAI121 00001870
      V      V      -PHAI12*DPHAI122*PHAI121*THR22 00001880
      V      V      -AC23=-(DPHAI12*WR32)*PHAI122*PHAI121 00001890
      V      V      -PHAI12*DPHAI122*PHAI121*THR31 00001880
      C-----*
      C      DO 1840 I=1,3
      C      C1840 AC3(I)=-(DPHAI1*(3)*WR(1,3))*PHAI12*(3)*PHAI12*(2)
      C      -PHAI1*(3)*DPHAI12*(3)*DPHAI12*(2)*THR0(1,3) 00001900
      C.TN   C.TN   -PHAI11*(3)*WR13)*PHAI123*PHAI122 00001910
      C.TN   C.TN   -AC31=-(DPHAI13*WR13)*PHAI123*PHAI122 00001890
      V      V      -PHAI13*DPHAI123*DPHAI122*THR13 00001900
      V      V      -AC32=-(DPHAI13*WR23)*PHAI123*PHAI122 00001890
      V      V      -PHAI13*DPHAI123*DPHAI122*THR22 00001900
      V      V      -AC33=-(DPHAI13*WR33)*PHAI123*PHAI122 00001890
      V      V      -PHAI13*DPHAI123*DPHAI122*THR32 00001900
      C-----*
      C      DO 1860 I=1,3
      C      C860 AC(I)=AC(1)+(AC1(I)+AC2(I)+AC3(I))*FACTOR
      V      V      AC(N0,1)=AC(N0,1)+(AC11+AC21+AC31)*FACTOR
      V      V      AC(N0,2)=AC(N0,2)+(AC12+AC22+AC32)*FACTOR
      V      V      AC(N0,3)=AC(N0,3)+(AC13+AC23+AC33)*FACTOR
      C-----* DEBUG *
      C.TN   WRITE(6,66665) <DPHAI1(I),I=1,3>, <PHAI1(I),I=1,3>,
      *      WRITE(6,66665) <DPHAI11,DPHAI12,DPHAI13,PHAI112,PHAI13>,
      C.TN   <DPHAI12,I),I=1,3>, <PHAI2(I),I=1,3>,
      *      DPHAI121,DPHAI122,DPHAI123,PHAI122,PHAI123>
      C.TN   <(THR0(I,J),I=1,3,J=1,3)>, <THR11,THR31,THR12,THR22,THR32,THR13,THR23,THR33>,
      C.TN   <AC1(I),I=1,3>,
      *      <AC11,AC12,AC13>,
      C.TN   <AC2(I),I=1,3>,
      *      <AC21,AC22,AC23>,
      C.TN   *
      C-----*
SSN=0434
SSN=0435
SSN=0436

```

Fig.5.7 Modified Subroutine GR241 (continued)

```

★ GR241 < 17 >

      FILE=J9127.ZGR12DVP.FORT77(GR241)  FORTAN 77/VP V12L1.0
      C.TN   *(AC3(C1),I=1,3)
      1     AC31,AC32,AC33
      1     WRITE(6,6666)  (AC(NO,I),I=1,3),
      1     (WR(C1,J),I=1,3), J=1,3)
      1     WR11,WR21,WR31,WR12,WR22,WR32,WR13,WR23,WR33
      1     *6665  FORMAT(1H ','* DEBUG ---- SUBROUTINE GR245 '*,
      1     *           /1H ','DPHA1(1---3)          ,3(IX,D23.16),
      1     *           /1H ','PHAI1(1---3)          ,3(IX,D23.16),
      1     *           /1H ','DPHA12(1---3)         ,3(IX,D23.16),
      1     *           /1H ','PHAI2(1---3)          ,3(IX,D23.16),
      1     *           /1H ','THRO(1---3,1)        ,3(IX,D23.16),
      1     *           /1H ','THRO(1---3,2)        ,3(IX,D23.16),
      1     *           /1H ','THRO(1---3,3)        ,3(IX,D23.16),
      1     *           /1H ','AC1(1---3)           ,3(IX,D23.16),
      1     *           /1H ','AC2(1---3)           ,3(IX,D23.16),
      1     *           /1H ','AC3(1---3)           ,3(IX,D23.16),
      1     *           /1H ','AC((1---3)          ,3(IX,D23.16),
      1     *           /1H ','WR(1---3,3)           ,3(IX,D23.16),
      1     *           /1H ','WR(1---3,3)           ,3(IX,D23.16),
      1     *           /1H ','WR(1---3,3)           ,3(IX,D23.16)
      1     C----- DEBUG.
      1     5900 CONTINUE
      1     2300 CONTINUE
      1     C.TN  I240 = 0
      1     V--4200 CONTINUE
      +     ENGPO = ENGPO + W1ENG + W2ENG
      1     RETURN
      1     END
      SSN=0437
      SSN=0438
      SSN=0439
      SSN=0440
      SSN=0441
      SSN=0442

```

Fig.5.7 Modified Subroutine GR241 (continued)

```
TWC(4 2 8 1 4) SRP  
//FORTEXVP EXEC FOLRTEXVP, SO='J0001.ZGR12DVP',  
//      A='ELM(*),NOS', OPT=F  
//GRL2      EXEC LKEDCTEX, LM='J0001.ZGR12DVP', UNIT='D0340'  
//
```

Fig.5.8 JCL for Making Load-module

```
TWC(6 4 8 1 4) SRP CLASS(0)  
//GR2GO EXEC LMGO='J0001.ZGR12DVP'  
//FT05F001 DD DSN=J0001.ZGRAPE.DATA(PART2), DISP=SHR  
//FT18F001 DD DSN=J0001.ZCR18E.DATA, DISP=SHR  
//FT19F001 DD DSN=J0001.ZCR19E.DATA, DISP=(NEW,CATLG),  
//      SPACE=(TRK,(100,100),RLSE), UNIT=TSSWK
```

Fig.5.9 JCL for Execution

6. おわりに

以上、情報システムセンター外来研究員が1993年度に行った原子力コードのベクトル化及び高速化についての作業内容及び作業結果を記した。今回扱ったコード（Cella を除くと）は、当研究所だけでなく多くの研究機関及び大学などで盛んに研究に用いられている典型的なタイプのシミュレーションコード（分子動力学、流体解析）である。これらのタイプのコードによる計算需要は今後も当研究所において大きな割合を占めるばかりでなく、計算機の発達に伴い一層、大きな系や長時間のシミュレーションが期待されるであろう。そのような状況下においては、スーパーコンピューターの性能を極限まで使用する利用技術の進歩が不可欠であり、情報システムセンターは今後も利用技術の研究及び開発に努力していく。

謝辞

コードのベクトル化及び改良に関して有益な技術指導をしてくださった情報システムセンター神林獎氏、及びコンピューター利用に関して常に適切な助言をしていただいている（株）富士通、相沢広氏に感謝いたします。また、これら作業を円滑に行うために各種の事務処理をしていただいた山田圭子氏に感謝いたします。

なお、（株）富士通 R & D システム部部長、橋本道夫氏には、常に外来研究員の各種研究作業に対し様々な形で援助していただきここに感謝いたします。

最後に本研究作業及び本レポートを書く機会を与えて下さいました情報システムセンター室長、秋元正幸氏に感謝いたします。

6. おわりに

以上、情報システムセンター外来研究員が1993年度に行った原子力コードのベクトル化及び高速化についての作業内容及び作業結果を記した。今回扱ったコード(Cellaを除くと)は、当研究所だけでなく多くの研究機関及び大学などで盛んに研究に用いられている典型的なタイプのシミュレーションコード(分子動力学、流体解析)である。これらのタイプのコードによる計算需要は今後も当研究所において大きな割合を占めるばかりでなく、計算機の発達に伴い一層、大きな系や長時間のシミュレーションが期待されるであろう。そのような状況下においては、スーパーコンピューターの性能を極限まで使用する利用技術の進歩が不可欠であり、情報システムセンターは今後も利用技術の研究及び開発に努力していく。

謝辞

コードのベクトル化及び改良に関して有益な技術指導をしてくださった情報システムセンター神林獎氏、及びコンピューター利用に関して常に適切な助言をしていただいている(株)富士通、相沢広氏に感謝いたします。また、これら作業を円滑に行うために各種の事務処理をしていただいた山田圭子氏に感謝いたします。

なお、(株)富士通R&Dシステム部部長、橋本道夫氏には、常に外来研究員の各種研究作業に対し様々な形で援助していただきここに感謝いたします。

最後に本研究作業及び本レポートを書く機会を与えて下さいました情報システムセンター室長、秋元正幸氏に感謝いたします。

参 考 文 献

- [1] 根本 俊行, 鈴木 孝一郎, 他; 原子力コードのベクトル化と改良, JAERI-M 92-142, 1992年9月.
- [2] 根本 俊行, 江口則地, 他; 原子力コードのベクトル化と改良(II), JAERI-M 92-142, 1993年7月.
- [3] 田次 邑吉; ダイヤモンド型と黒塩型結晶における照射損傷シミュレーションのための分子動力学コード: DGR 及び GGR, JAERI 1291, 1984年6月.
- [4] 小川 益郎; 黒塩酸化を伴なう多成分混合気体の物質伝達に関する数値解析, JAERI-M 92-139, 1992年10月.
- [5] 「核融合炉安全性解析コード—多成分固気相化学反応計算一」報告書, (財)原子力データセンター, 平成5年1月.
- [6] 「改良 STREAM コード—多成分・固気相化学反応計算一」ガイダンス, (財)原子力データセンター, 平成5年6月.
- [7] 「核融合炉安全性解析コード—多成分固気相化学反応計算(2)一」報告書, (財)原子力データセンター, 平成5年6月.
- [8] 「核融合炉安全性解析コード—多成分固気相化学反応計算(3)一」報告書, (財)原子力データセンター, 平成5年9月.
- [9] OS IV/F4 MSP FORTRAN77EX/VP 使用手引書 V12用, 富士通(株), 1991年7月.
- [10] OS IV/F4 MSP FORTRAN77EX 使用手引書 V12用, 富士通(株), 1991年7月.
- [11] OS IV/MSP アナライザ使用手引書 V10L20用, 富士通(株), 1991年8月.
- [12] 根本 俊行, 鈴木 孝一郎, 他; 新FORTRANコンバイラの導入とベクトル計算機の効果的利用法, JAERI-M 92-105, 1992年7月.