



Title	第一原理分子動力学シミュレーションコードのチューニング
Author(s)	広瀬, 喜久治; 後藤, 米子; 後藤, 英和 他
Citation	大阪大学大型計算機センターニュース. 1994, 91, p. 37-46
Version Type	VoR
URL	https://hdl.handle.net/11094/66043
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

第一原理分子動力学シミュレーションコードの チューニング

大阪国際女子大学 広瀬喜久治 後藤米子
京都工芸繊維大学 機械システム工学科 後藤英和
関西日本電気ソフトウェア 共通システム事業部 川口正仁
大阪大学 工学部 精密工学科 森 勇藏

1 はじめに

計算機の高速化と大容量化が進み、量子力学の第一原理に基づく分子動力学シミュレーション [1] を、固体材料などの多数の原子から成る系に対して適用することが可能になってきた。材料表面や内部における原子オーダーの現象に対する定量的信頼性の高いシミュレーションが実行可能であるため、材料を扱う研究分野において重要な研究手段として確立されつつあり、物性研究、材料設計、プロセス設計など、様々な分野での応用が試みられている [2][3][4][5]。

第一原理分子動力学シミュレーションを行うには、材料内部の電子状態を求める必要があるが、このとき大規模なエルミート行列（10000 × 10000 程度）と多数のベクトル（数百本）との積を実行しなければならない。今回、SX-3 用に開発した分子動力学シミュレーションプログラムでは、行列とベクトルの積が律速段階となっていることが判明し、この部分に対してチューニングを試みた結果、メモリと計算速度の両面で有効な方法が見いだせた。大規模な行列と多数のベクトルとの積は、他の研究分野においても頻繁に遭遇する計算であると考えられるため、今回行ったチューニング内容の詳細をここに報告する。

2 計算方法の要約

材料内部の電子状態を決定する波動関数 $\Psi_i(\mathbf{r})$ は、次式の *Kohn - Sham* 方程式と呼ばれる固有値方程式によって与えられる (*Hartree* 単位系) [6]。

$$\left[-\frac{1}{2} \Delta + V_{ion}(\mathbf{r}) + \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' + \mu_{xc}(\rho(\mathbf{r}))\right] \Psi_i(\mathbf{r}) = \varepsilon_i \Psi_i(\mathbf{r}) \quad (1)$$

ここに、 $\rho(\mathbf{r})$ は、位置 \mathbf{r} における電荷密度、 $V_{ion}(\mathbf{r})$ は原子核と内殻電子の与えるポテンシャル、 $\mu_{xc}(\rho(\mathbf{r}))$ は交換・相関ポテンシャル [7]、 $\Psi_i(\mathbf{r})$ は一電子波動関数、 ϵ_i はエネルギー固有値である。

$V_{ion}(\mathbf{r})$ としては、ノルム保存型擬ポテンシャル [8] と呼ばれるものを用いている。ここで、基底関数として平面波基底関数

$$\phi_{\mathbf{k}_n}(\mathbf{r}) = \frac{1}{\sqrt{N_0\Omega}} e^{i\mathbf{k}_n \cdot \mathbf{r}} \quad (\mathbf{k}_n \equiv \mathbf{k} + \mathbf{G}_n) \quad (2)$$

(N_0, Ω はそれぞれ単位胞の数と体積、 \mathbf{k}, \mathbf{G}_n はそれぞれブロッホ関数の波数ベクトルと逆格子ベクトル) を用い、

$$\Psi_i(\mathbf{r}) = \sum_{\mathbf{k}_n} C_{i,\mathbf{k}_n} \phi_{\mathbf{k}_n}(\mathbf{r}) \quad (3)$$

のように波動関数を展開すると、式 (1) は各波数ベクトル \mathbf{k} ごとの固有値方程式に分解され、

$$\begin{bmatrix} \frac{1}{2}\mathbf{k}_0^2 + \tilde{V}_0 & \tilde{V}_{-1} & \tilde{V}_{-2} & \cdots \\ \tilde{V}_1 & \frac{1}{2}\mathbf{k}_1^2 + \tilde{V}_0 & \tilde{V}_{-1} & \cdots \\ \tilde{V}_2 & \tilde{V}_1 & \frac{1}{2}\mathbf{k}_2^2 + \tilde{V}_0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} C_{i,\mathbf{k}_0} \\ C_{i,\mathbf{k}_1} \\ C_{i,\mathbf{k}_2} \\ \vdots \end{bmatrix} = \epsilon_{i,\mathbf{k}} \begin{bmatrix} C_{i,\mathbf{k}_0} \\ C_{i,\mathbf{k}_1} \\ C_{i,\mathbf{k}_2} \\ \vdots \end{bmatrix} \quad (4)$$

のように離散化される。

ここに、 \tilde{V}_j は、電子の受けるポテンシャル、

$$V(\mathbf{r}) = V_{ion}(\mathbf{r}) + \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' + \mu_{xc}(\rho(\mathbf{r})) \quad (5)$$

のフーリエ変換である。

この固有値方程式を解き、式 (3) の展開係数 C_{i,\mathbf{k}_n} を求めれば、電子状態が決定される。電子状態が決定されれば、系全体のエネルギーと各原子に働く力 (*Hellmann - Feynman* 力) を計算することができ [9]、それに従って各原子を移動させれば、いわゆる分子動力学シミュレーションを実行することができる。

固有値問題は、ハウスホルダー変換を用いて直接解くことも可能である。しかし、

Kohn - Sham 方程式の場合、行列要素は固有ベクトルに依存しているため、いわゆるセルフコンシステントな収束計算が必要になり、多大な計算時間を要する。そこで、記憶容量と収束性で有利な最急降下 (*Steepest Descent*) 法や、さらに収束性に優れた共役勾配 (*Conjugate Gradient*) 法を応用した計算方法が開発されている [10]。

今回チューニングしたプログラムは、最急降下法を応用したものである [3]。最急降下法とは、

$$(H - \epsilon_i)\Psi_i^{(0)} \simeq 0 \quad (6)$$

のように、固有値問題の解に近い初期固有ベクトル $\Psi_i^{(0)}$ を選び、

$$\Psi_i^{(n+1)} = \Psi_i^{(n)} + \mu \Delta \Psi_i^{(n)} \quad (7)$$

に従って、正しいベクトルに収束させてゆく方法である。ただし、 μ は加速パラメーター、 $\Delta \Psi_i^{(n)}$ は、

$$\Delta \Psi_i^{(n)} = -(H - \epsilon_i) \Psi_i^{(n)} \quad (8)$$

で表される残差ベクトルである。

分子動力学シミュレーションを行う場合、1ステップ前の固有ベクトルは、現在の時刻における固有ベクトルの良い近似になっていると考えられ、上記のような収束法が有利である。また、系内の電子数に応じた、必要最小限の本数の固有ベクトルについてだけ計算を行えること、更にハミルトニアン行列要素をあらかじめ記憶しておかなくても計算可能であることも有利な点である。

3 プログラムのアルゴリズムと問題点

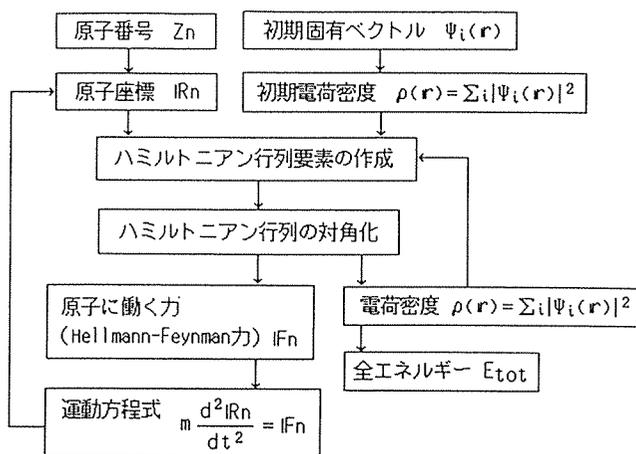


図 1: プログラムのフローチャート

図 1 に、プログラムのフローチャートを示す。原子座標と初期ベクトルを読み込んだ後、ハミルトニアン行列の対角化を行い、原子に働く力を計算し、原子座標を更新した後、プログラムの最初に戻る。

さて、ハミルトニアン行列の対角化を実行するためには、式 (8) を見てもわかるように、行列とベクトルの積を実行する必要がある。また、加速パラメーターなどを計算するときにも行列とベクトルの積が必要となる。更に、ここでは詳細は述べないが、今回のプログラムでは、最急降下法の収束を速めるためにレーリー・リッツの補正をおこなっており [3]、

そこでも行列とベクトルの積を計算している。また、原子に働く力を計算するときにも、行列とベクトルの積を実行する必要があり、プログラム全体を通してみると、4箇所で行列とベクトルの積を計算していることになる。そして、この部分が最も計算時間を消費しており、今回のチューニングでの検討箇所とした。更に、式(4)で示されているように、行列要素はフーリエ変換で求められるため、3次元複素データの高速フーリエ変換 (*FFT*) が必要である。従って、今回のチューニングでは、3次元 *FFT* サブルーチンの改良も行った。

4 チューニング内容

4.1 行列とベクトルの積

行列 H とベクトル $V1$ との積を計算し、ベクトル $V2$ に格納する以下のような操作について考える。

$$V2(K, J) = \sum_I [H(J, I) * V1(K, I)] \quad (9)$$

ただし、 K はベクトルの通し番号である。

チューニング前の *DO* ループ構造は以下のようなものである。

```

DO 10 K = 1, NV, 4   (ベクトルの走査)
DO 20 I = 2, NB     (行列の列の走査)
DO 30 J = 1, I - 1  (行列の行の走査：上三角のみ)
  行列要素  $H_{ji}$  の計算
   $V2(J, K) = V2(J, K) + H_{ji} * V1(K, I)$ 
  ( $K = K \sim K + 3$ )
30 CONTINUE
20 CONTINUE
10 CONTINUE

```

ここで、定数 NV は固有ベクトルの本数を表し、*DO 10* は固有ベクトルについて回るループである(ただし、4段のアンローリングが施してある)。また、 NB は基底関数の数、すなわち行列の行数(列数)を表し、*DO 20* と *DO 30* は、それぞれ行列の列を回るループと行を回るループである。ハミルトニアン行列はエルミート行列であるため、行を回すループでは、上三角行列に含まれる部分のみ回している。また、行列要素は記憶せず、ループ内で計算している。

この方法では、ループ構造の内側に最も大きい *DO* ループが配置されているが、行列要素の計算を固有ベクトルの本数の回数だけ行わねばならず、効率が悪い。この効率を上げるには、行列要素の計算を最初に1回だけ行っておき、記憶しておけばよいことになる。この場合の、ループ構造をタイプ1と名付け、以下に示す。

```

DO 10 I = 2, NB   (行列の列の走査)
  DO 20 J = 1, I - 1   (行列の行の走査：上三角のみ)
    行列要素  $H(J, I)$  の計算と配列へのストア
20 CONTINUE
10 CONTINUE
  DO 30 K = 1, NV, 4   (ベクトルの走査)
    DO 40 I = 2, NB   (行列の列の走査)
      DO 50 J = 1, I - 1   (行列の行の走査：上三角のみ)
         $V2(J, K) = V2(J, K) + H(J, I) * V1(K, I)$ 
          ( $K = K \sim K + 3$ )
50 CONTINUE
40 CONTINUE
30 CONTINUE

```

この方法の問題点は、行列要素の記憶のための大量の配列領域（ハミルトニアン行列の上三角）が必要となることである。そこで、今回のチューニングでは、ループの入れ替えを行い、タイプ2と名付けた。そのループ構造を以下に示す。

```

DO 10 I = 2, NB   (行列の列の走査)
  DO 20 J = 1, I - 1   (行列の行の走査：上三角のみ)
    行列要素  $H_{ji}$  の計算と配列  $H_i(J)$  へのストア
20 CONTINUE
  DO 30 K = 1, NV, 4   (ベクトルの走査)
    DO 40 J = 1, I - 1   (行列の行の走査：上三角のみ)
       $V2(J, K) = V2(J, K) + H_i(J) * V1(K, I)$ 
        ( $K = K \sim K + 3$ )
40 CONTINUE
30 CONTINUE
10 CONTINUE

```

このループ構造では、行列要素の計算を DO 20 で1行（上三角に含まれる部分のみ）だけ実行し、配列に記憶する。このようにすれば、行列要素計算の実行回数は、タイプ1と同じであり、しかも記憶用の配列は1行分で済むことになる。タイプ2のループ構造がタイプ1に比べてどの程度の性能を示すかを明らかにすることが、今回の実験の主要目的である。

4.2 3次元 FFTサブルーチン

3次元 FFTは、ASLライブラリの1次元 FFTを用いて、スキヤニング法で実行している。チューニング前は、初期化処理を含むDFCFBルーチンを使用していたが、これをSX対応のライブラリに置き換えた。即ち、初期化処理を行うJFC1SEと、変換処理を行うJFC1SFを利用した。

5 チューニングの効果

チューニングプログラムについて、テストパターンと中規模シミュレーションの2通りの計算を行い、チューニング前との比較を行った。テストパターンは、スーパーセル内に8個の原子を含むシリコン単結晶に対する計算を例にとり、 x, y, z 各方向の平面波基底関数の数を15個、固有ベクトルを16本とり、原子を移動させる分子動力学計算を行わず、電子状態の収束計算のみを行った。この場合、ハミルトニアン行列は 3375×3375 の複素エルミート行列となる

また、中規模シミュレーションでは、スーパーセル内に40個の原子を含むシリコン単結晶(001)表面に対する計算を例にとり、 x, y, z 各方向の平面波基底関数の数をそれぞれ15, 15, 19個、固有ベクトルを80本とり、1fsの時間ステップで原子を移動させる分子動力学計算を1ステップだけ実行した。この場合には、ハミルトニアン行列は 4275×4275 の複素エルミート行列となる

5.1 テストパターンでの効果

表1に、ANALYZERを用いてプログラム特性の調査を行った結果を示す。ただし、Tunedとは、タイプ1のプログラムに対する結果を示している。Mainプログラムはループ構造の改善で1.6倍、FFTルーチンで1.3倍、プログラム全体のトータルでは1.5倍の速度性能の向上がみられる。また、行列要素の計算に含まれていたINDEX処理が減少したことで、Bank Conflictが解消した。

3次元FFTサブルーチンでは、MOPSとV.Op.Ratioの値が低下しているが、これはベクトル化されていた不要な初期化処理が無くなったことによるもので、シミュレーションの規模が小さいことによる見かけの性能低下である。

表2に、プログラム全体に関するインフォメーションと、プログラム中で測定したCPU時間を示した。今回、行列とベクトルの積に対するループ構造の改良を行った、ハミルトニアン行列の対角化の部分で0.37倍、プログラム全体で0.65倍の計算時間の短縮化が達成されている。これは、Flop Countで0.7倍、V.Element Countで0.5倍、V.Inst Countで0.4倍という計算量の減少の効果が大きい。

以上の結果から、更に規模の大きな(扱う行列の大きい)計算を行う場合、今回のチューニングが大きな効果をもたらすことが予想された。

表1 テストプログラムに対するANALYZERの結果

Prog. unit	Original		Tuned	
	Main	FFT	Main	FFT
Frequency	1	54	1	54
CPUtime(sec)	22.32	4.87	13.90 (1/1.6)	3.78 (1/1.3)
CPUtime(%)	82.08	17.92	78.61	21.39
MOPS	1663.8	159.9	1405.8	106.9
MFLOPS	631.3	39.3	710.9	15.5
V. Op. Ratio	99.05	76.69	98.21	31.61
Aer. V. Len	235.6	16.3	233.2	55.3
Banc. Conf	Warning	No	Observe	No
Total CPUtime(sec)	27.20		17.68 (1/1.5)	
Total MFLOPS	525.27		562.16	

表2 テストプログラムのチューニング結果

Program type	Original	Tuned	Tune/Org
user time(sec)	27.02	17.59	0.65
sys time(sec)	0.58	0.71	1.22
vector time(sec)	13.71	4.67	0.34
MFlops	528.63	565.01	1.07
MOPS	1403.61	1133.93	0.81
V. op. R(%)	98.54	96.84	0.98
V. LEN	194.47	228.33	1.17
Inst. Count (M)	746	715	0.96
V. inst. Count (M)	192	84	0.44
V. Element Count (M)	37376	19314	0.52
FLOP Count (M)	14285	9938	0.70
行列の対角化(sec)	13.28	4.93	0.37

表3 中規模シミュレーションプログラム(タイプ1)
のチューニング結果

Program type	Original	Tuned	Tune/Org
user time(sec)	971.43	228.41	0.24
sys time(sec)	29.40	7.26	0.25
vector time(sec)	842.76	147.54	0.18
MFlops	1073.93	2086.46	1.94
MOPS	2487.20	3421.05	1.38
V. op. R(%)	99.60	99.28	1.00
V. LEN	237.92	239.42	1.01
Inst. Count(M)	19672	8845	0.45
V. inst. Count(M)	10115	3240	0.32
V. Element Count(M)	2406579	775812	0.32
FLOP Count(M)	1043245	476576	0.46
行列の対角化(sec)	154.53	38.66	0.25
力の計算(sec)	713.32	113.92	0.16
1 time step(sec)	848.81	157.89	0.19

表4 中規模シミュレーションプログラム(タイプ2)
のチューニング結果

Program type	Original	Tuned	Tune/Org
user time(sec)	971.43	236.55	0.24
sys time(sec)	29.40	9.10	0.31
vector time(sec)	842.76	160.79	0.19
MFlops	1073.93	2022.22	1.94
MOPS	2487.20	3718.20	1.38
V. op. R(%)	99.60	99.40	1.00
V. LEN	237.92	239.65	1.01
Inst. Count(M)	19672	8929	0.45
V. inst. Count(M)	10115	3648	0.36
V. Element Count(M)	2406579	874276	0.36
FLOP Count(M)	1043245	478366	0.46
行列の対角化(sec)	154.53	38.66	0.25
力の計算(sec)	713.32	113.92	0.16
1 time step(sec)	848.81	157.89	0.19

5.2 中規模シミュレーションでの効果

表3は、シリコン単結晶表面に対する実際のシミュレーションパラメーターを用いた場合の結果である。パラメーターが大きくなったことにより、チューニングを施した処理部分の比重が増大し、高速化の効果が非常に大きくなっている。プログラム全体の *Flops* 値で、約2倍の $2GFlops$ が得られた。これは、実アプリケーションとしては、非常に良い値である。

また、計算時間では、更に高速な0.24倍の短縮となっている。このなかで、行列とベクトルの積に対するチューニングを施した、行列対角化の部分では0.25倍、*Hellmann - Feynman*力の計算部分で0.16倍、そして分子動力学1ステップあたりの計算時間では0.19倍の短縮が達成されている。これらは、計算速度の向上と計算量の減少の相乗効果である。

表4は、タイプ2のプログラムに対する結果であり、タイプ1とほぼ同様の効果が得られていることがわかる。従って、行列と多数のベクトルとの積を計算する場合のループ構造に対して、行列要素をすべて記憶させるための大きな配列を用いなくても、用いた場合と同様の高速化を達成することが可能であることがわかった。

6 おわりに

今回のチューニングが成功したことで、比較的記憶容量の小さなジョブでも第一原理分子動力学シミュレーションが実行できるようになり、より効率的に $SX-3$ を利用することが可能となった。また、これまでは不可能であった大規模なシミュレーションを実行することも可能になり、シミュレーションの対象も拡大した。

今回のプログラムでは、電子状態の収束計算に最急降下法を用いたが、より収束性に優れる共役勾配法を用いる場合にも、同様なチューニングが可能である。

参考文献

- [1] R.Car and M.Parrinello : Unified Approach for Molecular Dynamics and Density-Functional Theory, *Phys. Rev. Lett.* 55, 22 (1985) 2471.
- [2] 米沢富美子, 坂本昇一 : 第一原理分子動力学法による水素化アモルファスシリコンのシミュレーション, *応用物理*, 60, 10 (1991) 984.
- [3] 宇田 毅, 井原茂男 : $Si(001)$ 表面の分子動力学シミュレーション, *固体物理*, 25, 11 (1990) 867.
- [4] 広瀬喜久治, 後藤英和, 土屋八郎, 森 勇藏, 遠藤勝義, 山内和人 : 材料表面現象の第一原理分子動力学シミュレーション—シリコン単結晶 (001) 表面の水素終端化反応—, *精密工学会誌*, 掲載予定.

- [5] 広瀬喜久治, 後藤英和, 土屋八郎, 森 勇藏, 遠藤勝義, 山内和人: 水分子と相互作用するシリコン単結晶 (001) 水素終端化表面の第一原理分子動力学シミュレーション, 精密工学会誌, 投稿中.
- [6] W.Kohn and L.J.Sham : Self-consistent equations including exchange and correlation effects, Phys. Rev. 140, 4A (1965) 1133.
- [7] J.P.Perdew and A.Zunger : Self-interaction correction to density-functional approximations for many-electron systems, Phy. Rev. B23-10 (1981) 5048.
- [8] G.B.Bachelet, D.R.Hamann and M.Schluter : Pseudopotentials that work: From H to Pu, Phy. Rev. B26-6 (1979) 4199.
- [9] M.T.Yin and M.L.Cohen : Theory of lattice-dynamical properties of solids : Application to Si and Ge, Phy. Rev. B26-6 (1982) 3259.
- [10] I.Stich,R.Car,M.Parrinello and S.Baroni : Conjugate gradient minimization of the energy functional : A new method for electronic structure calculation, Phy. Rev. B39-8 (1989) 4997.