

7点差分用反復行列解法における前処理の効果

吉田有一郎、岡田悟、川田猛（東芝CAEシステムズ）、
吉田浩俊（東芝）

デバイスシミュレーションの3次元移流拡散方程式から生じる差分方程式を、不完全LDU分解の前処理付きBCG法、CGS法、BCGSTAB法と、スーパーコンピュータCRAY YMPを用いて解き、反復数・計算時間を検討した。不完全LDU分解のフィルインは有り無しを計算した。BCG法のみは、LDU行列の作用のさせ方も3種類計算した。例題はn型シリコンの電子密度分布解析である。格子数 $40 \times 40 \times 40$ 、 $100 \times 100 \times 100$ の2例題の計算の結果、以下の結論を得た。(1)フィルインの有る不完全LDU分解により反復回数は低減する。計算時間は反復数・格子数とも多い場合のみ短縮する。(2)LDU行列は行列Aの右側から作用させる場合が計算時間的に最も有利である。

Effect of Pre-conditioning on Iterative Method
for 7-Points Difference Matrix

Matrices generated from three dimensional drift diffusion equation using 7-Points difference have been solved by BCG, CGS, and BCGSTAB, pre-conditioned by incomplete LDU decomposition with and without fill-in. For BCG, three different usages of LDU matrix have been also considered. The electron distribution analyses of N type silicon cube with $40 \times 40 \times 40$ and $100 \times 100 \times 100$ grids using supercomputer CRAY Y-MP have shown that, (1) LDU decomposition with fill-in has less iterations, but, has less CPU time only in cases of large grids and many iterations. (2) LDU matrix applied to A matrix from the right took the least CPU time.

1. 緒言

半導体素子の電気的特性を計算するツールの一つとしてデバイスシミュレータがある。デバイスシミュレーション^{(1)・(2)}では、半導体素子中の電位、電子密度、正孔密度に関する偏微分方程式を解く。流体モデルを使用する場合、偏微分方程式は移流拡散方程式となる。離散化の手法は、差分法を使用する例が多い。半導体素子の設計最適化のためデバイスシミュレーションを実施する場合、解析は相当数繰り返される。したがって、作業時間短縮・計算機コスト低減のため、デバイスシミュレータは十分高速化することが必要である。特に、3次元計算では、膨大な計算をスーパーコンピュータによって高速に処理することが不可欠である。著者らの経験では、係数行列の発生部がベクトル処理により高速化されていれば、3次元計算で必要な計算時間の70%程度は、連立1次方程式の解法に費やされる。したがって、連立1次方程式の高速解法は、3次元デバイスシミュレーションでは特に強く要求される。

一般に、偏微分方程式を差分法により離散化して発生する大型の連立1次方程式を解く場合、使用するメモリ量、計算時間の点で、対称行列ではICCG法、非対称行列ではILUBCG法、ILUCGS法、ILUCR法などの前処理付き反復解法が有利なことが文献^{(3)・(4)・(5)}に示されている。前処理の不完全LDU分解に関する研究としては、Meijerink⁽⁶⁾、Gustafsson⁽⁷⁾、後⁽⁸⁾、Meijerink・van der Vorst⁽⁹⁾、佐藤ら⁽¹⁰⁾など、多くの研究がある。文献⁽⁹⁾、⁽¹⁰⁾では、2次元問題における、フィルインを増やした不完全LDU分解の効果を検討している。3次元問題に関しても、長谷川・村田⁽¹¹⁾により不完全LDU分解の有効性が示されている。しかるに、3次元問題の場合、フィルインを増やした不完全LDU分解の方法は、Meijerink・van der Vorstが文献⁽⁹⁾でその方法を示しているのみであり、実際の数値計算により有効性を検討している文献は見あたらない。そこで著者らは、Meijerink・van der Vorstらが提唱している不完全LDU分解の手法をBCG法、CGS法、BCGSTAB法⁽¹²⁾

に適用し、反復数・計算時間について検討を行なった。また、BCG法に関しては、LDU行列を作用させる方法についても併せて検討した。本報においては、その結果を報告する。

2. 前処理法

非対称行列Aを不完全LDU分解する方法は式(1)で与えられる。分解されるL, Uの形を設定すれば、

$$\begin{aligned} Ax &= b \\ A &= LDU - R \end{aligned} \quad \text{--- (1)}$$

式(2)を使用してL, D, Uの各項が求まり、Rが決まる。L, Uの形の設定方法は、文献⁽⁷⁾の中で

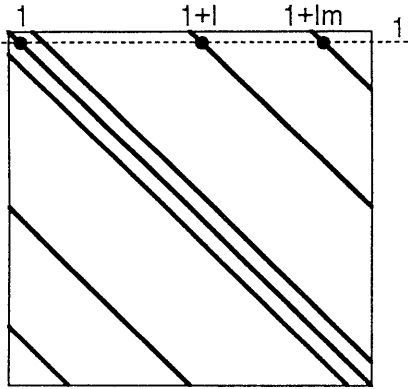
$$\begin{aligned} l_{ij} &= a_{ij} - \sum_{k=1}^{j-1} l_{ik} d_k u_{kj} \quad (j=1, i-1) \\ u_{ii} &= a_{ii} - \sum_{k=1}^{i-1} l_{ik} d_k u_{kj} \quad (d_i = u_{ii}^{-1}) \\ u_{ij} &= a_{ij} - \sum_{k=1}^{i-1} l_{ik} d_k u_{kj} \quad (j=i+1, n) \end{aligned} \quad \text{--- (2)}$$

Gustafssonが示している方法を使用する。この方法を以下に示す。すなわち、

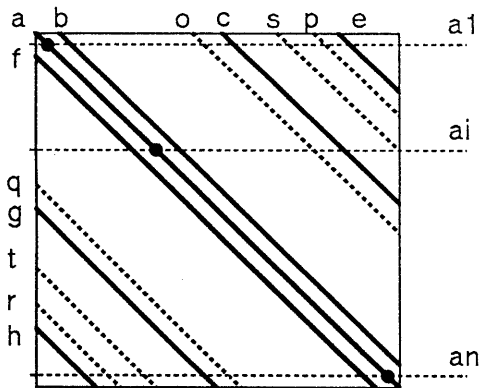
- (a) LとUとして、Aと同じ下三角成分、上三角成分を持つものを選ぶ
- (b) LとUの積を作り、Aの非ゼロ成分の位置と異なる、新しい非ゼロ成分がでてくる位置を求める。
- (c) 新たに非ゼロ成分が出てくる位置(フィルイン)と元の非ゼロ成分の位置を両方含めた形でL, Uの形を設定する。
- (d) (a)に戻る

(a)~(d)を繰り返すことにより、フィルインの多いL, Uの形を求めることができる。しかし、実際は、フィルインを増やしたLDU分解をすると演算量が増えるため、(c)の段階で新たに出てくるフィルインは対角項に吸収させ、非ゼロ項は増やさないという考え方がGustafssonの修正法⁽⁷⁾、後の修正法⁽⁸⁾の考え方である。Meijerink・van der Vorst⁽⁹⁾は(a)~(d)の方法により、最高33本の非ゼロ要素のある不完全LDU分解(ICCG^(7,9))を

使用している。彼らは、3次元問題の7点差分行列に(a)~(d)の方法を適用した場合に導かれる不完全LDU分解の形も示した。本報告で著者らが使用した不完全LDU分解はこれと同じものである。図1(a)に7点差分行列を、図1(b)にL, Uの非ゼロ成分の位置を示す。i行目の非ゼロ成分を図1(b)のように表わす場合、本報告で使用した不完全LDU分解は式(3)で与えられる。式(3)では、分解後



(a) 7点差分行列



(b) L, Uの非ゼロ成分の位置
(破線部がフィルイン)

図1 7点差分行列とL, Uの位置

の成分に波線をつけている。本報告では、比較のために、フィルインのない不完全LDU分解についても計算を行なった。フィルインのない場合の不完

$$\begin{aligned}
 \tilde{h}_i &= h_i \\
 \tilde{r}_i &= -h_i \tilde{d}_{i-lm} \tilde{b}_{i-lm} \\
 \tilde{t}_i &= -h_i \tilde{d}_{i-lm} \tilde{c}_{i-lm} - \tilde{r}_i \tilde{d}_{i-lm+1} \tilde{o}_{i-lm+1} \\
 \tilde{g}_i &= g_i - h_i \tilde{d}_{i-lm} \tilde{s}_{i-lm} \\
 \tilde{q}_i &= -\tilde{g}_i \tilde{d}_{i-l} \tilde{b}_{i-l} - \tilde{r}_i \tilde{d}_{i-lm+1} \tilde{s}_{i-lm+1} \\
 \tilde{f}_i &= f_i - h_i \tilde{d}_{i-lm} \tilde{p}_{i-lm} - \tilde{g}_i \tilde{d}_{i-l} \tilde{o}_{i-l} \\
 \tilde{a}_i &= a_i - \tilde{f}_i \tilde{d}_{i-1} \tilde{b}_{i-1} - \tilde{q}_i \tilde{d}_{i-1+1} \tilde{o}_{i-1+1} \\
 &\quad - \tilde{g}_i \tilde{d}_{i-1} \tilde{c}_{i-1} - \tilde{t}_i \tilde{d}_{i-lm+1} \tilde{s}_{i-lm+1} \\
 &\quad - h_i \tilde{d}_{i-lm} \tilde{e}_{i-lm} - \tilde{r}_i \tilde{d}_{i-lm+1} \tilde{p}_{i-lm+1} \\
 \tilde{b}_i &= b_i - \tilde{q}_i \tilde{d}_{i-1+1} \tilde{c}_{i-1+1} \\
 &\quad - \tilde{r}_i \tilde{d}_{i-lm+1} \tilde{e}_{i-lm+1} \\
 \tilde{o}_i &= -f_i \tilde{d}_{i-1} \tilde{c}_{i-1} - \tilde{t}_i \tilde{d}_{i-lm+1} \tilde{p}_{i-lm+1} \\
 \tilde{c}_i &= c_i - \tilde{t}_i \tilde{d}_{i-lm+1} \tilde{e}_{i-lm+1} \\
 \tilde{s}_i &= -\tilde{q}_i \tilde{d}_{i-1+1} \tilde{p}_{i-1+1} - \tilde{e}_{i-l} \tilde{d}_{i-l} \tilde{g}_i \\
 \tilde{p}_i &= -\tilde{f}_i \tilde{d}_{i-1} \tilde{e}_{i-1} \\
 \tilde{e}_i &= e_i
 \end{aligned} \tag{3}$$

全LDU分解は、式(3)で、

$$\tilde{r}_i = \tilde{t}_i = \tilde{q}_i = \tilde{o}_i = \tilde{s}_i = \tilde{p}_i = 0$$

として得られる。LDUによる前処理は式(4)、式(5)、式(6)の3通りの方法が考えられる。デバイスシミュレーションでは、式(4)を使用している例⁽¹³⁾、式(5)を使用している例⁽¹⁴⁾、式(6)を使用している例⁽¹⁵⁾のいずれの事例もある。本報告では、式(6)を使用したが、BCG法に関しては式(4)、式(5)についても計算を行ない結果を比較した。

$$A(LDU)^{-1}(LDU)x = b \tag{4}$$

$$(LDU)^{-1}Ax = (LDU)^{-1}b \tag{5}$$

$$L^{-1}A(DU)^{-1}(DU)x = L^{-1}b \tag{6}$$

式(4)~式(6)を使用した反復法の計算では、L, Uに関する前進・後退代入が必要である。スーパ

コンピュータを使用して高速演算を行なうためには、この部分をベクトル化することが必須である。この部分をベクトル化するためには超平面法を使用する。格子の番号付けをx軸、y軸、z軸の順に行なう場合、フィルインがない場合の超平面は式(7)で、フィルインが有る場合の超平面は式(8)で与えられる(x, y, z, k, k'は整数である)。

$$x + y + z = k \quad (7)$$

$$x + 2y + 3z = k' \quad (8)$$

3. 差分方程式

デバイスシミュレーションでは、未知数として、電位 ψ 、電子密度 n 、正孔密度 p を扱う。ここでは、電子密度についての支配方程式を式(9)、式(10)に示す。ここで、 e は電子の電荷、 GR_n は電子の生成率、 D_n は拡散係数、 J_n は電子電流密度ベクトル、 μ_n は移動度である。式(10)は移流拡散方程式と呼ばれる形の方程式である。1次元問題を考え、生成率を0、電流密度を定数とすれば、式(10)は式(9)を満たす。さらに、

$$\text{div} J_n + eGR_n = 0 \quad (9)$$

$$J_n = eD_n \text{grad} n - e\mu_n n \text{grad} \psi \quad (10)$$

式(10)は積分することができる。積分区間 $[0, \Delta x]$ で、電子密度を n_1, n_2 と与えると、電子密度の厳密解は式(11)で与えられる。

$$n = n_1 + \frac{1 - \exp(\mu_n \frac{d\psi}{dx} \Delta x / D_n)}{1 - \exp(\mu_n \frac{d\psi}{dx} \Delta x / D_n)} (n_2 - n_1) \quad (11)$$

また、式(10)より、電流密度に関する式(12)が得られる。式(12)は、Scharfetter-Gummelの式と呼ばれる。

$$J_n = e\mu_n \frac{\Delta \psi}{\Delta x} \frac{\exp(\mu_n \frac{d\psi}{dx} \Delta x / D_n) n_1 - n_2}{1 - \exp(\mu_n \frac{d\psi}{dx} \Delta x / D_n)} \quad (12)$$

式(12)を使用して導いた差分式は、厳密解を使用しているため、電界が強く移流項が支配的な場合から、電界が弱く拡散項が支配的な場合まで、安定な解を求めることができる。格子間の垂直2等

分面6枚から構成される直方体(コントロールボリューム)のx, y, z軸方向に式(12)を適用し、コントロールボリュームにおける電流の連続式を考えれば、7点差分の差分式が得られる。

4. 解析対象

図2に示すn型シリコン立方体中の電子密度分布を解析した。不純物濃度、材料定数は一様とした。電界はx軸方向にのみ一様な値を与えた。この場合、電界と電子密度のカップリングがないため、式(12)より導かれる差分式は電子密度に関して線形な方程式となる。立方体の3辺をx, y, z軸とし、x軸に垂直な2つの面上で電子密度(入口で 2×10^{10} 、出口で 1×10^{10})を与え、他の面上では電子電流の出入りはないものとした。電界の値を変化させ、移流項と拡散項の強さの比を変え、収束特性の変化を見た。出入口の電位差、

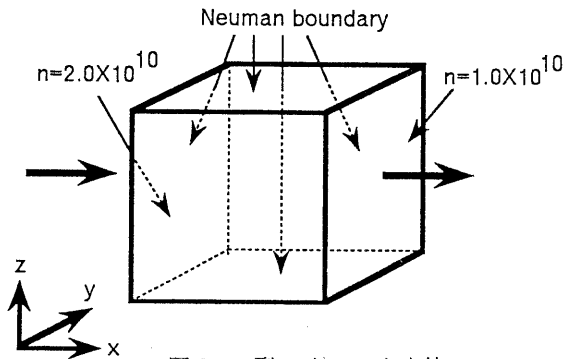


図2 n型シリコン立方体

拡散定数、および、移動度を用い、セルベクレ数は式(13)で表わされる。アインシュタインの関係、式(14)より、拡散定数は移動度と関係づけられている。式(14)中、 k はボルツマンの定数、 T は絶対温度である。本報告では、 $kT/e=1.0$ とおいた。この場合、セルベクレ数はセル中の電位差と

$$P = \mu_n \frac{d\psi}{dx} \Delta x / D_n \quad (13)$$

$$D_n = \frac{kT}{e} \mu_n \quad (14)$$

等しくなる。格子数は $40 \times 40 \times 40$ (64000点)、および、 $100 \times 100 \times 100$ (1000000点)とした。収束判定は、

前処理された方程式の残差ベクトルの自乗ノルムと前処理された方程式の右辺ベクトルの自乗ノルムの比の平方根 ϵ を使用し、許容値 $\epsilon \leq 10^{-9}$ とした。計算には CRAY-YMP4128 と CFT77 を使用した。計算は 64bit 計算である。

5. 結果と考察

図3に、格子数64000点の場合における、反復数とセルペクレ数を示す。図3では、フィルインのある不完全LDU分解はフィルインのない不完全LDU分解よりも、反復数が低減している。低減率はBCG法で29%~30%、CGS法で28%~40%、BCGSTAB法で26%~37%である。セルペクレ数が0.025より大きい範囲における反復数の低減効果はBCG法よりもCGS法、BCGSTAB法が大きい。

図4に、格子数1000000点の場合における、反復数とセルペクレ数を示す。図4では、フィルインのある不完全LDU分解の反復数の低減率は、BCG法で24%~58%、CGS法で25%~33%、BCGSTAB法で26%~38%である。CGS法の場合、フィルインのない不完全LDU分解の場合は収束しないが、フィルインのある不完全LDU分解の場合は収束する例がある。セルペクレ数が0.05より大きい範囲における反復数の低減効果はBCGSTAB法よりもBCG法、CGS法が大きい。

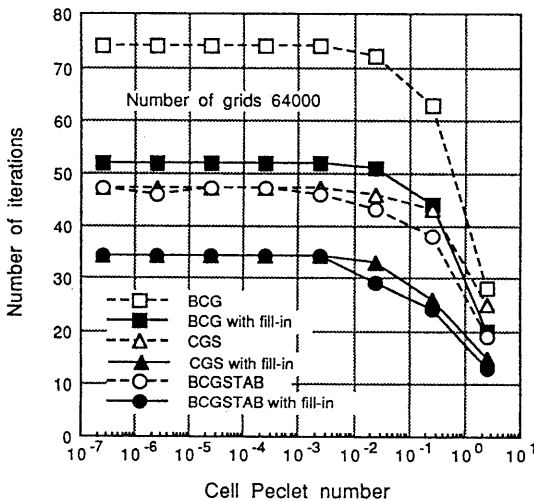


図3 反復数とセルペクレ数 (格子数64000点)

図5に、格子数64000点の場合における、計算時間とセルペクレ数を示す。図5では、フィルインのある不完全LDU分解により、計算時間が短縮している場合と、増加している場合がある。全体的には、フィルインのある不完全LDU分解により計算時間が増加する場合が多い。

図6に、格子数1000000点の場合における、計算時間とセルペクレ数を示す。図6でもやはり、フィルインのある不完全LDU分解により、計算時間が短縮している場合と、増加している場合がある。

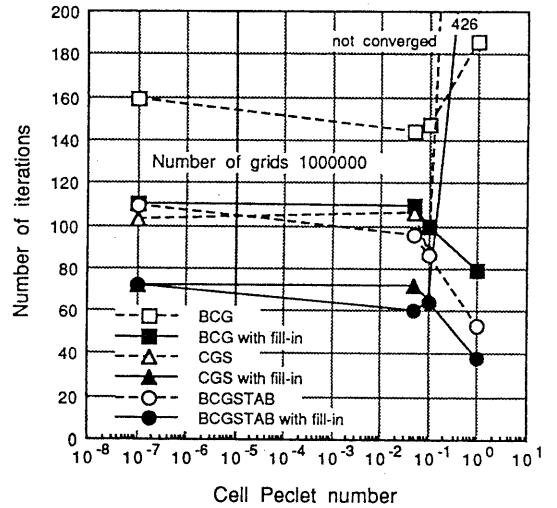


図4 反復数とセルペクレ数 (格子数1000000点)

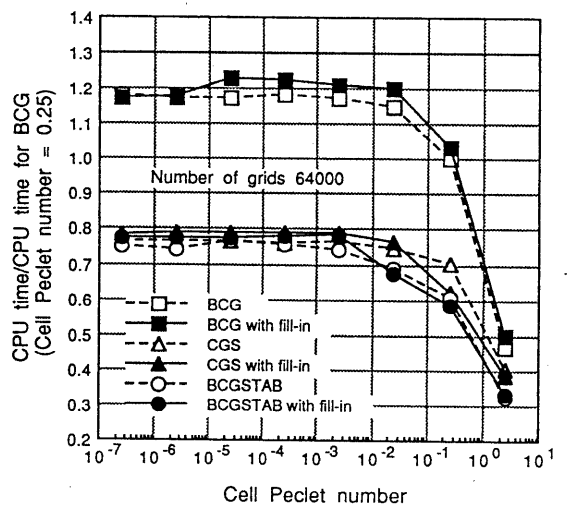


図5 計算時間とセルペクレ数 (格子数64000点)

全体的には、反復数が多くかかる場合は、不完全LDU分解により計算時間が減少する傾向にある。

図7に、LDU行列の作用のさせ方を変えたBCG法（格子数64000点）の反復数とセルペクレ数を示す。図7では、フィルインのある不完全LDU分解による反復数の低減効果は、行列Aの右側からLDU行列を作用させた場合が最も大きく、低減率は25%~35%である。

図8に、LDU行列の作用のさせ方を変えたBCG法（格子数1000000点）の反復数とセルペクレ数を示

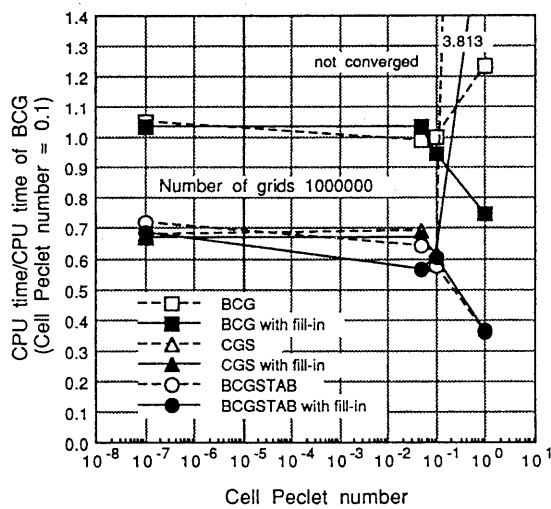


図6 計算時間とセルペクレ数（格子数1000000点）

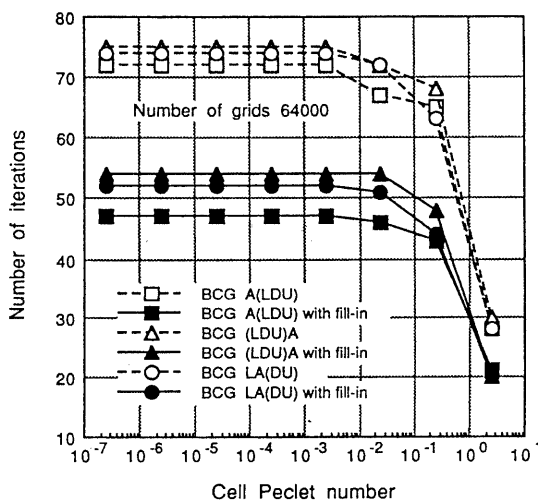


図7 反復数とセルペクレ数
（BCG法、格子数64000点）

す。図8では、フィルインのある不完全LDU分解による反復数の低減効果は、セルペクレ数が0.05より大きい範囲においては、行列AをL行列とDU行列で挟む場合が最も大きく、低減率は24%~58%である。

図9に、LDU行列の作用のさせ方を変えたBCG法（格子数64000点）の計算時間とセルペクレ数を示す。図9では、計算時間の短縮効果は、行列Aの右側からLDU行列を作用させた場合が最も大きく、短縮率は-11%~7%である。全体的には、計算時間

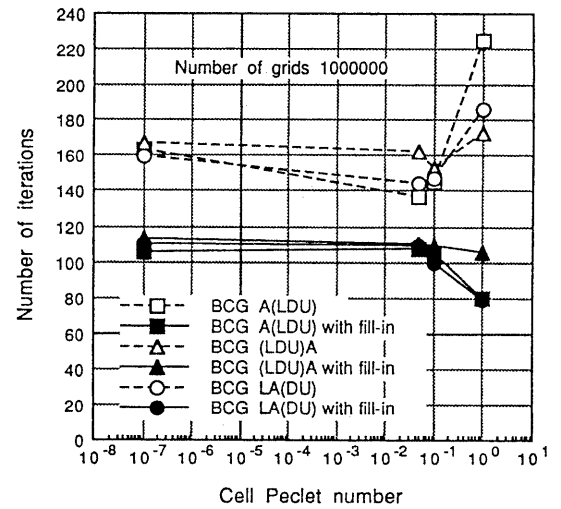


図8 反復数とセルペクレ数

（BCG法、格子数1000000点）

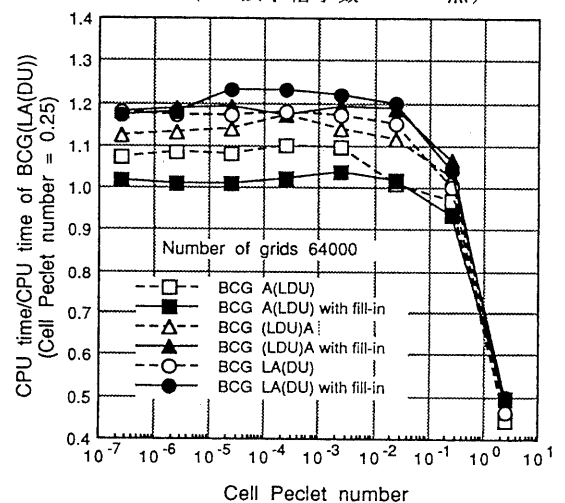


図9 計算時間とセルペクレ数
（BCG法、格子数64000点）

の短縮効果はあまりない。

図10に、LDU行列の作用のさせ方を変えたBCG法（格子数1000000）の計算時間とセルペクレ数を示す。図10では、計算時間の短縮効果は、行列Aの右側からLDU行列を作用させた場合が最も大きく、短縮率は-10%~49%である。全体的にも計算時間が短縮している場合が多く、セルペクレ数が0.1より大きい場合は計算時間の短縮効果が大きい。

図11に、格子数1000000、セルペクレ数1の場合のBCG法の残差と反復回数を示す。式(4)は式(6)

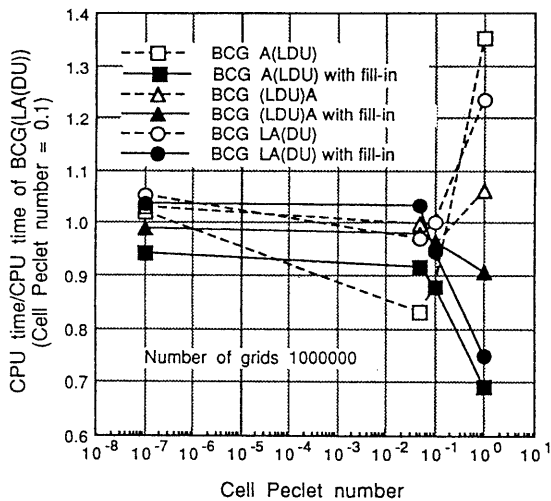


図10 計算時間とセルペクレ数
(BCG法、格子数1000000点)

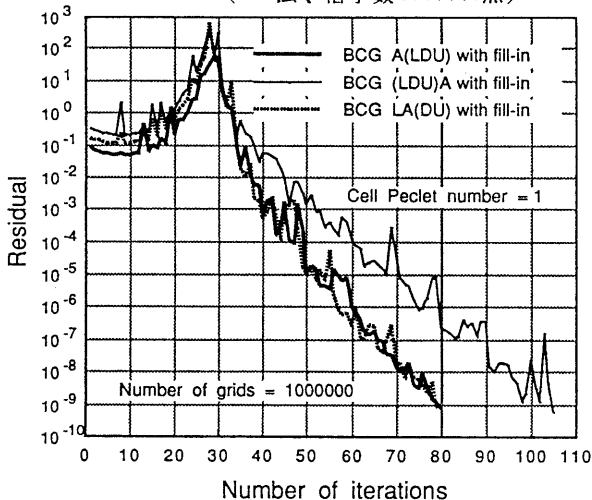


図11 残差と反復回数
(BCG法、格子数1000000点)

に近い収束の様子を示している。

式(5)、式(6)では、前処理後の差分方程式の残差ベクトルのディメンジョンと、元の差分方程式の右辺ベクトルのディメンジョンが異なる。したがって、式(4)、式(5)、式(6)では、前処理後の差分方程式の残差ベクトルの減少と元の差分方程式の収束が最も良く対応しているのは式(4)と考えられる。係数行列の性質が悪くなった場合、残差ベクトルの減少と元の差分方程式の収束の対応に差が出るのが予想されるが、本報告の例題では、式(4)、式(5)、式(6)ともに解の精度に差はなかった。

フィルインのある不完全LDU分解による計算時間の短縮効果については、使用計算機がスーパーコンピュータのようなベクトル計算機であるか、汎用計算機のようなスカラ計算機であるかにより差が生じる。特にスーパーコンピュータの場合は、問題の格子数によってはメモリバンクコンフリクトが発生し、計算時間が増加する⁽¹⁶⁾。さらに、実際のデバイスシミュレーションでは、非線形問題をNewton Raphson法により解くが、強い非線形性のため、いきなり最終電圧に対する解を得ることはできない。電圧を増分的に与え、中間の電圧に対する解を求めながら方程式を解く。この増分の取り方によって係数行列の解き難さは変化する。増分を小さくすれば、増分計算の回数は増えるが、発生する行列は解き易くなる。増分を大きくすれば、増分計算の回数は減るが、行列は解き難くなる⁽¹⁰⁾。したがって、デバイスシミュレーションでは、反復過程の安定した行列解法は、反復過程が不安定な行列解法よりも増分を大きくとることにより、計算時間的に有利になる可能性がある。今後、以上の点を含めて検討を続けたい。

6. 結言

3次元の移流拡散方程式を7点差分により離散化して得られる行列を、不完全LDU分解の前処理付き反復行列解法（BCG法、CGS法、BCGSTAB法）により解き、反復回数・計算時間を検討した。不完全LDU分解のフィルインは有り無しを計算した。

BCG法については、LDU行列の作用のさせ方も3種類計算した。不完全LDU分解はMeijerink・Gustafssonが示した方法を使用した。解析した例題は、n型シリコン立方体中の電子密度分布解析である。格子数は64000点および1000000点である。スーパーコンピュータCRAY Y-MP4128を使用した計算の結果、以下の結論が得られた。

- (1) BCG法、CGS法、BCGSTAB法ともに、フィルインのある不完全LDU分解はフィルインのない不完全LDU分解よりも反復数が低減する場合が多い。
- (2) 計算時間の短縮に関しては、格子数が多い問題、反復数が多く必要な問題は、フィルインのある不完全LDU分解が有利となる場合があるが、格子数が少ない問題、少ない反復数で解ける問題に関してはかえって不利になる場合が多い。
- (3) CGS法に関しては、フィルインのない不完全LDU分解によって解けない問題が、フィルインのある不完全LDU分解で解ける場合がある。
- (4) BCG法における前処理行列の作用のさせ方は、LDU行列を行列Aの右側から作用させる場合が、計算時間的に最も有利である。

文 献

- (1) 檀編, プロセス・デバイスシミュレーション技術, (1988), 産業図書.
- (2) 倉田, バイポーラトランジスタの動作理論, (1980), 近代科学社.
- (3) 村田・ほか2名, スーパーコンピュータ, (1985), 丸善.
- (4) 名取・野寺編, スーパーコンピュータと大型計算, bit臨時増刊, 11(1987), 共立出版.
- (5) 三好編, コンピュートロール, Vol. 26, (1989), コロナ社.
- (6) Meijerink, J., A., and, van der Vorst, H., A., Math. of Comp., 31-137(1977), 148.
- (7) Gustafsson, I., BIT, 18(1978), 142.
- (8) 村田・ほか2名, 大型数値シミュレーション, (1990), 岩波書店.

- (9) Meijerink, J., A., and, van der Vorst, H., A., J. Comp. Physics, 44(1981), 134.
- (10) 佐藤・ほか2名, 電子通信学会論文誌, J69-CN11, (1986), 1389.
- (11) 長谷川・村田, 文献(4), 52.
- (12) van der Vorst, A., SIAM J. Sci. Stat. Compt., to appear.
- (13) Heiser, G., Claude, P., Weis, J., and Fichtner, W., IEEE Trans. Computer-Aided Design, CAD10-10(1991), 1218.
- (14) 鳥谷部, 文献(4), 1812.
- (15) Zhao, Z., Zhang, Q., Tan, G., and Xu, J., M., IEEE Trans. Computer-Aided Design, CAD10-11, (1991), 1432.
- (16) 藤野・長谷川, 日本応用数学会論文誌, 2-2(1992), 105.