

解説

スーパーコンピュータと
デバイスシミュレーション†

羽根 邦夫†

1. はじめに

近年の半導体集積回路 (IC) に使用されている半導体素子 (トランジスタ) は非常に微細化されている。半導体素子の微細化により IC に含まれる素子の数が増えるばかりでなく、高速化、高信頼化、低消費電力化などにも効果的である。微細化により生じる電氣的な問題点を解決するためと、優れた特性が得られるように、素子の構造は年々複雑となっている。このため、素子特性の解析や設計は複雑となり、計算機解析 (シミュレーション) を導入することが必須になった。

さらに、IC の開発に計算機解析を導入することにより、試作の回数が大幅に減少し、開発に要する費用と時間は大幅に減少する。このため、計算機による設計 (CAD) および開発の技術はここ 5 年ほどで急激に発達し、広く実用化されている。

半導体の開発に用いられる半導体シミュレータは大きく分けて三つある。まず「プロセスシミュレータ」は半導体素子の製造条件から、素子の形状と内部構造を詳細に計算する。「デバイスシミュレータ」は形状と内部構造から、素子内部の電位、電荷分布や、外部端子の電氣的特性を計算する。「回路シミュレータ」は回路の配線と抵抗、容量、トランジスタの特性から回路の応答を計算する。これらのシミュレータを結合すれば IC の設計を効率的に行えるわけである。計算機利用と数値解析の立場からみれば、プロセス、デバイス、回路と、いずれも興味深い¹⁾。

従来はこれらのシミュレータ用のソフトウェアは米国製が多かったが、1983 年に米国政府はこれらソフトウェアのソースレベルでの輸出を禁止した。以来我が国の主力半導体メーカーは自前でこれらのシミュレータを作っている。

ここでは主として流体型デバイスシミュレータを対象として、高精度、高速な数値計算方法を述べ、スーパーコンピュータの有効利用方法についても述べる。従来のデバイスシミュレータにはガウス消去法、SOR 法、SIP 法などの行列計算方法を用いたものが多かったが、ここでは最近注目されている高速、高精度な共役勾配法 Conjugate Gradient Method (CG) 系の行列計算方法を用い、計算方法の最適化について説明をする。

ガウス消去法はメモリ使用量が多く、行列の元数が大きくなると丸め誤差の蓄積が問題となる。SOR 法は大行列では収束に要する反復回数が加速度的に増加し、計算を繰り返しても誤差が減少しなくなることもある。SIP 法は収束に要する反復回数が多く、収束速度を高めるパラメータの最適化が必要なため、実用的でない。

これに対して、CG 系の行列計算法はメモリ使用量も少なく、前処理を行うことにより収束速度を高めることが可能である。また、逐次的に近似解を修正するように計算するので、誤差も少なくできる。

従来の計算方法では、15 MFLOPS の汎用スカラ計算機で、一つの動作点 (電圧条件) あたりの計算時間は数分以上かかっていた。素子の特性をつかむには数百点の計算を要するから、総計算時間は十時間以上を必要とする。

集積回路開発にはさまざまな構造の素子をテストすることになるから、コンピュータ資源の利用と集積回路開発の立場からは、これでは計算時間がかかり過ぎる。デバイスシミュレータの計算量のなかで大半を占めるのが行列方程式を解く部分で、全計算量の 90% 以上である。そこで行列計算をいかに高速化するかが重要である。

ここではまずスカラ計算機を用いて高速、高精度な行列計算方法を試行し、その上で、スーパーコンピュータを使い、短時間で計算する方法を論じる。

† Semiconductor Device Simulation Code for Super-computer by Kunio HANE (Department of Electrical Engineering, Faculty of Science and Technology, Keio University).

†† 慶應義塾大学理工学部電気工学科

2. 基礎方程式

デバイスシミュレータは、与えられたトランジスタの構造と動作条件のもとに、内部の電圧、電流、電子密度、正孔密度などを計算する。シミュレータの使用により、現状の技術では測定することが不可能な部分や瞬間的なできごとでも詳細に観察することができる。

デバイスシミュレータには移動する電荷の挙動を粒子レベルで計算するモンテカルロ型もある。これはこれでスーパーコンピュータ向きの計算であるが、まず流体型デバイスシミュレータについて述べる。

解析する空間を2次元として基礎方程式を以下に示す。ポアソン方程式は素子内部の電位分布を計算する。この方程式は1元2階の偏微分方程式であり、楕

円型方程式に分類される。

$$\nabla \cdot (\epsilon \nabla \phi) = -q(N_d - N_a + p - n) \quad (1)$$

ここで、 ϕ は電位、 N_d と N_a は不純物密度、 n は電子密度、 p は正孔密度である。 ϵ は誘電率、 q は素電荷である。

電流連続方程式は素子内部密度を計算する。

$$\nabla \cdot J_n = q(R - G) \quad (2n)$$

$$\nabla \cdot J_p = -q(R - G) \quad (2p)$$

ここで、(2)式の電流密度 J は以下のようになり、半導体内部の電流担体である電子の密度 n と正孔の密度 p を用いて表現される。

$$J_n = -q(\mu_n n \nabla \phi - D_n \nabla n) \quad (3n)$$

$$J_p = -q(\mu_p p \nabla \phi + D_p \nabla p) \quad (3p)$$

μ は電子あるいは正孔の移動度であり、電界および電荷密度の関数である。これに電界 $\nabla \phi$ を掛けて $\mu \nabla \phi$ は速度となる。(3)式を代入すると、(2)式は2元2階の偏微分方程式となり、この方程式は放物型方程式に分類される。

デバイスシミュレータは上述のポアソンおよび電流連続方程式を連立(結合)させ、両方の方程式が満足する電位 ϕ 、電子密度 n 、正孔密度 p 、などの分布を求める。

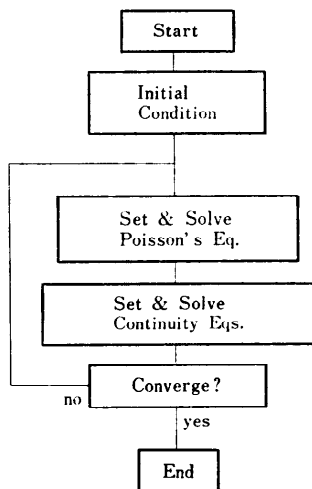
ポアソン方程式と電流連続方程式の結合方法として二つの方法がある。二つの方程式を別個に解く、Decoupled 法と、同時に解く Coupled 法である。両者のアルゴリズムを図-1(a)、(b)に示す。

3. 離散化

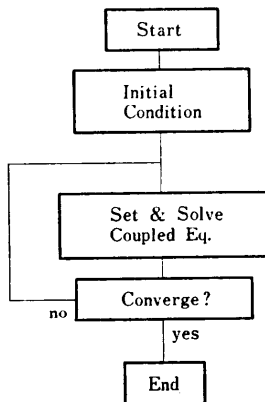
基礎方程式の離散化には長方形メッシュを用いた有限差分法を例とする。離散化方法には、このほかに三角形や四、六角形メッシュを用い、複雑な空間を近似するのが容易な有限要素法や、任意形状の四角形メッシュを用いる Boundary-Fit 方式の有限差分法がある。

トランジスタの製造工程をシミュレートするプロセスシミュレータもかつては有限差分法のシミュレータが多かった。最近は微細化にともない素子形状が複雑になり、形状適合性が優れた有限要素法、移動境界型や Boundary-Fit 方式の有限差分法を使用する例もある。

デバイスシミュレータではいまだに長方形メッシュを用いた有限差分法が多い。簡単で、精度、計算時間の点で満足できるためである。解析できる領域は長方形あるいは単純な斜線で表現できるものまでである。幸いにも電気的特性に影響をもつ領域は特定できるの



(a) Decoupled 法のアルゴリズム



(b) Coupled 法のアルゴリズム

図-1

で、長方形で区切れる適当な領域を対象とし、適切な境界条件を使えば、いまのところは差分法でも使用できる。図-2 にプロセスシミュレーションの領域（実線）とデバイスシミュレーションの領域（点線）を示す。

しかし、今後超高集積素子を解析する場合、トレンチや LOCOS などの複雑な構造も解析領域に含まれるであろう。このため、形状適合性の高い有限要素法の計算方法にもふれる。

まず、有限差分法について述べる。ここではトランジスタ内部を図-3のように分割し、離散的に電荷密度や電位分布などの物理量を与える。実線の交点には点線で囲まれた領域を代表する密度や電位などのスカラー量を与える。点線は領域の境界であり、電界や電流などのベクトル量を与える。解析ではこれらの量を計算する。

離散化するにあたって、離散化誤差を低減させ、電流と電束の保存則が成り立つよう工夫 (Gauss の定理に基づく Box integration) する。また、計算量の低減のために、各物理量は規格化して無次元とする。Box integration の概念は、(1)、(2)式の Divergence を計算するにあたって図-3 の点線部分での流れ (Flux) を線積分し、点線で囲まれた領域への出入り (Divergence) を計算する。

[Decoupled 法の離散化]

ポアソン方程式は右辺の n, p が電位 ϕ の関数で

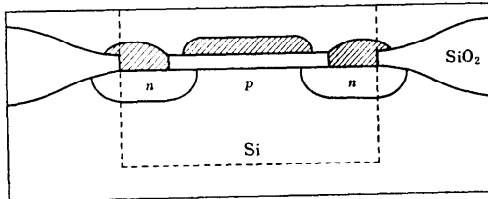


図-2 MOS トランジスタを例とした素子構造。点線はデバイスシミュレータ、実線はプロセスシミュレータの対象とする領域

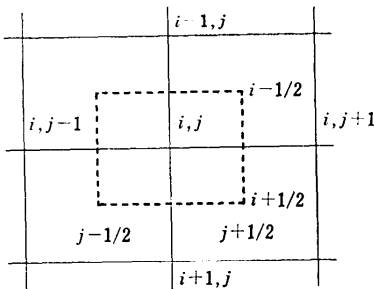


図-3 有限差分法のグリッド

あるため非線形であり、ニュートン法を用いて線形化する。すなわち、電子密度 n と正孔密度 p は電位 ϕ の関数であるから、 ϕ の微小変化に対する n と p の変化分は、以下のように導ける。 n を例として、

$$n = n_i \exp\left(\frac{q}{kT}(\phi - \phi_n)\right) \quad (4)$$

$$\frac{\partial n}{\partial \phi} = \frac{q}{kT} n \quad (5)$$

とする。ここで n_i は真性キャリア密度、 ϕ_n は電子の擬フェルミレベルである。 p についても同様に変化分を求めて、ポアソン方程式に代入すると、(1)式は

$$\begin{aligned} \nabla^2 \phi - \frac{q^2}{\epsilon kT} (p + n) \delta \phi \\ = -\frac{q}{\epsilon} (N_A - N_D + p - n) - \nabla^2 \phi \end{aligned} \quad (6)$$

となる。この式を $\delta \phi$ を未知数として、テイラー展開による有限差分法で離散化する。これによりポアソン方程式は線形連立方程式となり、行列形式で表せる。形成される行列は対称正定値行列である。

電流連続方程式を離散化する場合、テイラー展開を用いると移流項が存在するため、収束性が非常に悪い。そこで、Gummel-Sharfetter²⁾ らが提案した方法で流れを保存しながら風上化を行う。電子密度と、正孔密度にニュートン法を適用して、電流連続式を以下のように変形する。

$$\begin{aligned} \frac{1}{q} \frac{\delta J_{ni+1/2} - \delta J_{ni-1/2}}{(\Delta x_{i+1} + \Delta x_{i-1})/2} + \delta G - \delta R \\ = \frac{1}{q} \frac{J_{ni+1/2} - J_{ni-1/2}}{(\Delta x_{i+1} + \Delta x_{i-1})/2} - G - R \end{aligned} \quad (7)$$

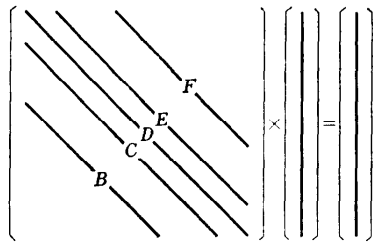
ここで、 $J_{ni+1/2}$ と $J_{ni-1/2}$ は図-3 の点線上的の電流密度である。さらに、(3)式の電流密度式にアインシュタインの関係式を適用して、 δJ_n をもとめると、

$$\begin{aligned} \delta J_{ni-1/2} &= \frac{\partial J_{ni-1/2}}{\partial n_{i-1}} \delta n_{i-1} + \frac{\partial J_{ni-1/2}}{\partial n_i} \delta n_i \\ &= q \mu_{i-1/2} \frac{\phi_{i-1} - \phi_i}{\Delta x_i} \left\{ \frac{1}{1 - \exp\left(\frac{q}{kT}(\phi_{i-1} - \phi_i)\right)} \right\} \delta n_{i-1} \\ &\quad + q \mu_{i-1/2} \frac{\phi_{i-1} - \phi_i}{\Delta x_i} \left\{ \frac{1}{1 - \exp\left(\frac{-q}{kT}(\phi_{i-1} - \phi_i)\right)} \right\} \delta n_i \end{aligned} \quad (8)$$

これにより、安定に収束させることができる。電流連続方程式から形成される行列は非対称非正定値となる。

これらの方程式はともに以下のような形式の行列となる。

ここで、 A は正方行列で、元数はメッシュの数となる。

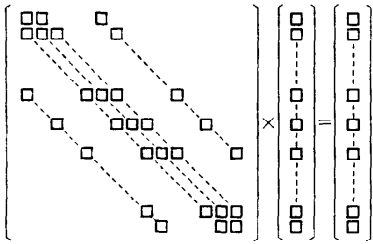


$$Ax=b$$

(9)

【Coupled 法の離散化】

Coupled 法は二つの方程式を連立させて解き、ポアソン方程式と電流連続方程式とともにニュートン法により線形化する。離散化したグリッドごとに電位、電子密度、正孔密度を表す 3×3 のブロック小行列を作り、一回の行列計算で解く。この結果形成される行列は非対称ブロック行列となり、以下のような形式の行列となる。



$$Ax=b$$

(10)

4. 境界条件

境界条件として、電極部は電位、電荷密度ともにディリクレ条件を適用した。電極以外の境界では、電位、電荷密度ともにノイマン条件を用いる。MOSFET においては、Si-SiO₂ 境界面が素子内部に存在するため、この部分では電束連続の条件を満足するようにする。このような境界条件を、有限差分法あるいは有限要素法に用いるのは容易である。

しかし、特殊な境界条件として、表面再結合や、ショットキ電極などの効果を考慮するには、数学的な工夫が必要である。このような場合、境界値が境界と内部の状態で決まるから、境界条件はどうしても不安定となる。

有限要素法ではこの場合、境界条件を離散化段階で組み込み、行列計算で一括して計算できるので簡単に処置できる。差分法では境界値のみを別個に計算し、この値を行列計算の境界値として挿入するので面倒である。

5. 行列計算方法

5.1 規則的スパース行列

有限差分法で形成される行列は規則的スパース行列である。このような行列で非零要素に注目すると、要素は斜めのベクトル状に並んでいる。この形式を利用して、2次元の場合は5本（5点差分）、3次元の場合は7本のベクトルで行列を表現し、計算する。これにより、零要素を無視でき、メモリ使用量を大幅に低減できる。

このように必要なデータのみを使って、行列を表現しても、行列解法にガウス消去法などの直接解法を使用しては、三角化分解時に多量のフィルインが発生し、メモリ使用量の低減にはならない。そこで、共役勾配法 (CG) 系の行列解法を用いて、少ないメモリ使用量で高速に計算する。

1952 年に CG 法が発表されたとき³⁾、ガウス消去法を凌ぐ計算方法として大反響を呼んだ。しかし、実際には思ったほど計算は早くならず、正定値行列しか解けないため、最近までかえりみられることがなかった。CG 系の計算方法が広く用いられるようになったのは、前処理が行われるようになってからである⁴⁾。

前処理の導入により、CG 法は直接解法をメモリ使用量、計算速度の面で凌いだ。以来、CG 法系の行列解法は各方面で研究され、正定値が保証されない、弱い非対称行列を解くことが可能な解法まで開発されている。現在ガウス消去法の優れている点は、よほどの悪条件行列でなければ解ける点である。一方、CG 系の計算方法はパイプライン型スーパーコンピュータにも適しており、今後ますます使用されることであろう。図-1(a)の Decoupled 法を例として、前処理の概念を簡単に述べよう。

ポアソン方程式あるいは電流連続方程式を離散化して得た線形連立方程式を $Ax=b$ とする。ここで A を三角化分解する。ただし、分解は不完全な形式で、

$$A=M+N \quad (11)$$

とする。ここで M は A の近似行列であり、 N は誤差分である。

不完全三角化分解とは完全分解の場合に (9) 式の B と C 、 E と F の間に発生する要素（フィルイン）を無視することで、 N は無視された要素に対応する。

実際には A が対称行列の場合は不完全コレスキー分解 (IC 分解) により、

$$A=LDL^T+N \quad (12)$$

とする。ここで L は下三角行列、 D は対角行列、 L^T は L の転置行列である。 A が非対称行列の場合は不完全 LU 分解 (ILU 分解) により、

$$A = LU + N \quad (13)$$

とする。ここで U は上三角行列である。

次にこれを方程式に作用させる。

$$M^{-1}Ax = M^{-1}b \quad (14)$$

$M^{-1}A$ を一つの行列とすればこの行列は単位行列に近く、固有値は 1 付近に密集する。この行列方程式を CG 系の解法で解く。CG 系の行列解法は収束に要する反復回数が行列の固有値分布に影響され、固有値が密集していれば反復回数は少なくなる。

A に対する M の作用のさせかたは、対称行列の場合は、

$$(LD^{1/2})^{-1}A(D^{1/2}L^T)^{-1}D^{1/2}L^Tx = (LD^{1/2})^{-1}b \quad (15)$$

とする⁴⁾。非対称行列の場合は、

$$(LU)^{-1}Ax = (LU)^{-1}b \quad (16)$$

とする。

このようにして前処理を行うが、三角化分解が不完全であるために前処理の効果が不十分であることが多い。対称行列の前処理を例にとると、IC 分解において、 L と L^T の要素数を下に示すように A と同一とする場合を、IC(1, 1) と呼ぶ。前処理としては簡単な方法である。しかし、デバイスシミュレータでは行列の次元数が数千を越えることも珍しくなく、このような大行列に対して IC(1, 1) では、 LDL^T の近似度が不十分でこの逆行列を A に作用させても固有値の密集化は不十分である。

そこで、より強力な前処理を施して誤差成分行列の N を減少させ、近似行列 LDL^T の近似度を向上させる。これには二つの方法がある。一つは要素数増加型で、以下に示すように IC 分解のときにフィルインを一部だけ考慮して、 L と L^T の要素数を増す方法である。

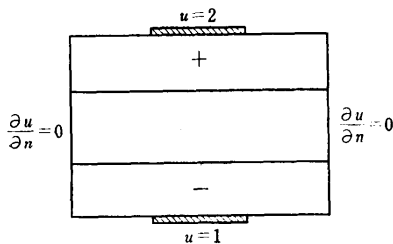
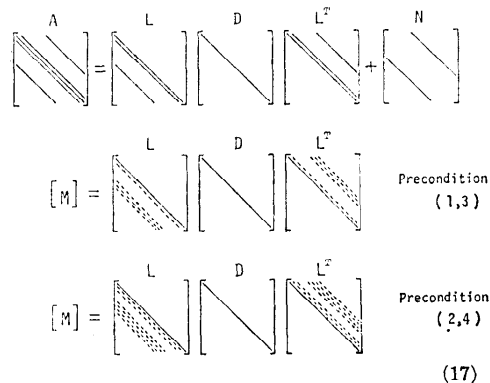


図-4 行列計算方法テスト用素子モデル



ここで (2, 4) などの数字は非対角ベクトルの本数を示し、要素数を増せば増すほど近似行列の精度が向上し、収束に必要な反復回数は減少する。しかし、1 反復あたりの計算時間は逆に増加し、メモリ使用量も多くなる。

近似行列の近似度を上げる他の方法としてはグスタフソンの Modification⁵⁾がある。これについても試行する。

前処理の効果をみるために、図-4に示すような 2 次元に電荷が分布するモデルを 40×40 のメッシュで離散化して 1600 次元の連立方程式からなるテスト行列を作った。図中の u は変数を表し、+ は正電荷、- は負電荷を表す。

この連立方程式を前処理方法を変えて CG 法で解

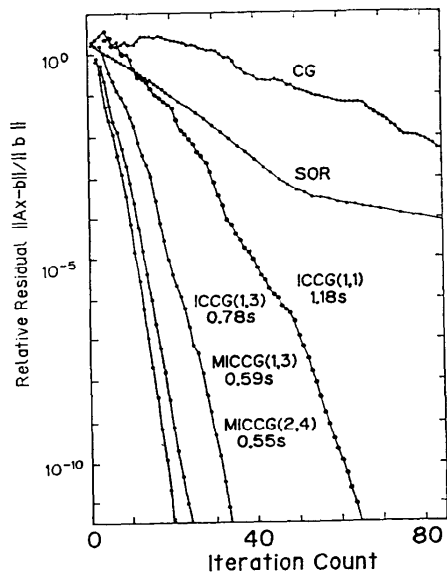


図-5 対称行列計算方法の比較と前処理の効果

き、収束の状況を残差ノルムに対する反復回数と、収束（ノルム $< 10^{-12}$ ）に要する計算時間の関係を図-5に示す。計算時間は HITAC M280H を IAP を用いない状態で使用し、倍精度計算をした結果であり、残差ノルムはユークリッドノルムを用いた。少ない反復回数で収束する場合は、前処理に用いた近似行列の精度が高く固有値の密集化がより良く行われ、収束性が向上したといえる。

図中のM記号は Modification を施したことを意味する。なお、前処理なしの CG と SOR は比較のために示したもので、明らかに SOR は収束性に難がある。

図-6 は各種の前処理方法の効果を、収束に要する反復回数と計算時間によって比較したものである。Modification には最適な収束パラメータ σ を使用しており、 L と L^T の要素数の増加が少ないときに顕著な効果を示し、MICCG (1, 3) では、一段上の ICCG (2, 4) に近い性能となった。しかし、ICCG (2, 4) と MICCG (2, 4) を比較すると、Modification の効果はほとんどなく、要素数増加のみでも十分な効果が得られた。

一方、電流連続方程式を離散化した線形連立方程式から形成される規則的スパース非対称非正定値行列の計算方法として、高速、高精度な計算方法として、CG 的な ORTHOMIN(q) (OR) (Conjugate Residual Method, CR 法とも呼ぶ)、BCG⁶⁾、CG-Squared Method (CGS)⁷⁾ を比較する。これらには、いずれも前処理として不完全 LU 分解 (ILU 分解) およびグスタフソンの Modification を用いた。テスト行列は図-5 で示したものと同一である。

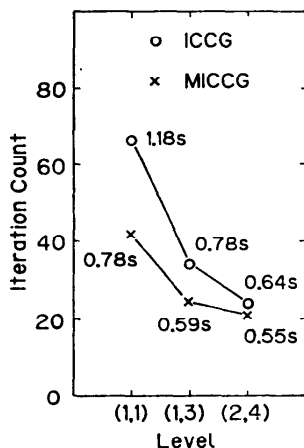


図-6 前処理の効果

この結果、ILU (1, 1) は他の方法と比較して反復回数が大きく、収束性に難点がある。Modification は (1, 3) レベルまでは有効であった。(2, 4) レベル以上では、最適な σ を用いても反復回数の低減に顕著な効果がみられず、要素数の増加のみで十分である。計算時間の点では、簡単な ILU (2, 4) が短かった。

次に三つの行列計算方法を同じ行列により比較してみる。前処理は ILU (2, 4) を用いており、図-7 にその結果を示す。この結果、収束性、計算時間の点で CGS が最も優れているといえる⁸⁾。

以上は図-1(a)に示す Decoupled 法の場合であったが、ポアソン方程式と電流連続方程式を一括して解く Decoupled 法の場合は、形成される行列は(10)式のような形式となる。

図-1(b)は Coupled 法のアルゴリズムを示すもので、ポアソン方程式と電流連続方程式を同時に解く。Decoupled 法の内部ループが3組あったのに対して、Coupled 法では一つしかなく、外部ループは同様の一つである。3×3のブロック小行列を一つの要素と考えれば、2次元モデルの場合は要素がベクトル状に5本並んだ規則的スパース行列となり、Decoupled 法で用いたと同種の行列計算方法が使用できる。(10)式で与えられたブロック行列は非対称であるため、行列計算方法は非対称行列計算用の ILUCGS をもとにした、Block-ILUCGS (BILUCGS) を使用する。

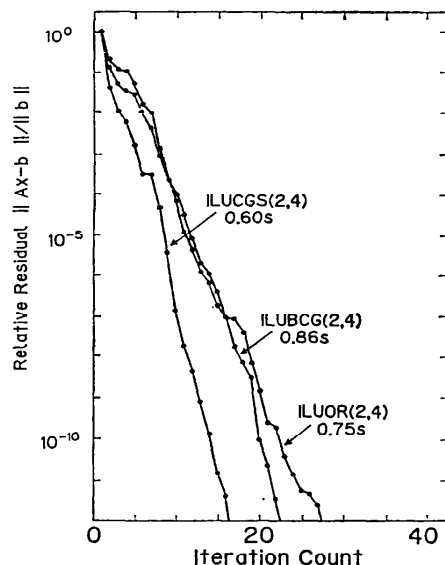


図-7 非対称行列計算方法の比較

前処理も Decoupled 法と同様に要素数を増加させることにより、LU の近似度を高めることにする。しかし、要素数の増加は内部ループ反復回数を減らす。反面、一反復あたりの計算量とメモリ使用量の増加をひきおこす。メモリ使用量が限度を越えればページ書換えが起こり、計算時間は極端に増加する。DS にとって、このページ書換えの発生は極力避けねばならない。特に、Coupled 法の場合は、各ブロックには9つの値があるため (3×3 ブロック小行列)、Decoupled 法よりもはるかに多くのメモリを必要とする。このため、要素数の増加による前処理には注意を要する。

Coupled 法ではポアソン方程式と電流連続方程式を同時に解くため、実際のデバイスシミュレータ上で比較した。まず、BILUCGS (1, 1) で解いたところ、計算時間の点で不満足であるため、(1, 2)、(1, 3) と、要素数を増した。この結果、(1, 2) の前処理が計算処理とメモリ使用量で優れていた。発散の危険性については (1, 2)、(1, 3) とともに同程度であり、最終的には DS には BILUCGS (1, 2) が適当であるといえる。

5.2 不規則スパース行列

有限要素法で形成される不規則スパース行列は、行列計算の立場からみると非常に扱いにくく、計算方法も数少ない。差分法の場合規則的スパース行列で、簡単に非零要素をベクトルとして取り扱え、メモリを圧縮し、CG系の行列計算方法を適用できた。これに対して不規則スパース行列では非零要素がバラバラに存在し、このままではメモリ使用量が増加し、非零要素まで計算して計算量が膨大になる。

最近ではガウス消去用のベクトル化計算のアルゴリズムも準備され、スーパーコンピュータを用いるかぎりには、計算時間は問題とならない。しかし、半導体シミュレータの行列の元数は数万を超えることが珍しくなく、元数の二乗以上で増加する条件数を考えると、直接法では精度の点で不満が生じる。膨大なフィルインによるメモリ量と、ページ書換えも問題である。

そこで、自前でリストベクトルを作り、非零要素のみを蓄え、CG系の計算方法を適用する。これにより、メモリ使用量は大幅に減少し、計算精度も向上させることができる。

行列計算方法はポアソン方程式に対しては ICCG (1, 1) を、電流連続方程式に対しては ILDUCGS (1, 1) を使用する。前処理段階での要素数増加は不

規則行列なので、今のところはいへない。

ポアソン方程式においては、対角要素で他の要素を規格化したスケーリング前処理も有効である。IC 前処理では要素数増加ができないので、前処理が弱く、反復回数が多い。一方、スケーリング前処理では、さらに前処理のレベルは低いが、100% ベクトル化が可能である。このため、スーパーコンピュータ上では、IC 前処理とほとんど同じ計算時間である。スカラ計算機ではスケーリング前処理は固有値が密集化しないため、反復回数が多く、計算時間の点で使いものにならない。

5.3 ベクトル化

スーパーコンピュータを効率よく使用するには、スカラ計算機で用いた計算手法をそのまま転用せず、スーパーコンピュータ用の行列計算方法とベクトル化率を高め、使用するスーパーコンピュータの構成 (アーキテクチャ) に合わせてチューニングする。

これまでは行列計算方法の最適化を行ってきたが、スーパーコンピュータ上で計算を行うと別の問題が生じる。スーパーコンピュータを効率的に動かせるために、まずプログラムのベクトル化の作業を行うが、どこまでベクトル化ができるかが重要である。

ベクトル化とはパイプライン型計算機に適するようにプログラムを構成することで、最内側ループの繰返し回数をベクトル長と呼び、これをできるかぎり大きくすることと言える⁹⁾。現在の S810, SX2, VP などの並列パイプライン型スーパーコンピュータはこの考え方でプログラムを最適化する。

しかし、ベクトル化可能なのは最内側ループのみであるため、ループが重複すると、外側のループはベクトル化ができず、ベクトル化率は低下する。すなわち、2次元データはそのままでは効率よく計算できないので、1次元にデータ構造を変える。また、通常はパイプラインの立ち上がりに時間を要するので、ベクトル長が4以下ではスカラ計算よりもかえって遅く、数百を越えなければ十分な加速ができない。

有限要素法の不規則スパース行列ではベクトル化をしても、平均ベクトル長は2次元で10程度、3次元で20以下であり、加速は十分でない。したがって、有限要素法で現状のベクトル計算機を使用するのは効率的ではない。今後の研究が必要である。

【規則的スパース行列のベクトル化】

有限差分行列で、前処理とCG系の行列計算部分をベクトル化する場合、一番ベクトル計算がしにくいのは

は、(11)式から(13)式の前進消去と、(14)式から(16)式までの前進、後退代入の、要素を横に参照する部分であり、いずれも前処理のルーチンである。なんの工夫もしないで代入計算を行うと、要素数増加型の前処理の場合、再帰的なデータ参照が行われてベクトル計算ができなくなる。これらのルーチンでは、リストベクトル（間接指標配列）を使用し、グリッドの番号付けを変え、データ配列に工夫を加えて、データを連続的に参照するようにして、ベクトル計算をする。

CG系の行列計算部分では、ベクトル化は容易である。(9)式にあるB～Fの斜めの要素をベクトルと考えれば、ベクトル長は元数とほぼ同じで、数千以上になる。CGの計算アルゴリズムは三角化分解のように要素を横に参照することがないので、斜めにデータ参照をするようにプログラムすれば容易に加速率を上げることが可能である。

Coupled 法は、前に述べたように、通常の前処理(BILU)と行列計算(CGS)を行い、最後に 3×3 の小行列を展開してベクトル化して解く。

以上のような考え方に従って、ベクトル化を行うが、現状ではベクトル化の作業は多分にハード依存型で、かつ経験的な要素が強い。5.1 までが理論的あるいは実測により、比較、検討が可能であったのに対し、ベクトル化はノウハウの領域の話といえる。

また、5.1 までの行列計算方法の最適化がそのままベクトル計算機に適用できるものでもない。むしろ、有限要素行列に対するスケーリング前処理のように、レベルの低い前処理の方がベクトル化が容易であり、かつベクトル長が長くて加速率も大きく、計算時間が短くなることがある。特に対称行列の場合はこの傾向があるが、非対称性が強い行列の場合は、前処理のレベルが低いと行列の条件が悪くなり収束できなくなる。

以上の計算方法を S810/20 で実行した。この結果、前処理レベルの高い方が計算時間は短かった。スカラ計算に対するベクトル計算の計算時間の短縮率、すなわち、加速率はグリッド数の増加にともなって向上した。DSをスタートしてから終了するまでのトータルの加速率は、1000～3000グリッドの場合で、5～8であった。これは行列計算部分にベクトル長が短いルーチンが残っていること、行列計算部分以外はベクトル化しにくいことが原因である。

したがって、今後これ以上の加速をするために、離散化方法を含めて前処理、行列計算を計算機の構成を考慮しながら検討する必要がある。前述のようなベク

トル化の作業はハード依存性が強く、いまだ各種計算機で比較、検討していない。最近の CRAY-XMP や ALLIANT-FX/8 タイプの並列ベクトル型計算機、ハイパーキューブや PAX 型計算機も考慮に入れなければならない。結論を述べることはまだできない。

6. モンテカルロ法

モンテカルロ法は移動する荷電粒子（電子、正孔）の挙動を追跡する。半導体の棒を対象として、シミュレートする空間を無限に広く、一様の電界と仮定する。これにより、粒子の移動度、拡散係数、エネルギー分布を正確に計算することができる。デバイスシミュレータとして用いる場合は素子の形状と境界条件を設定し、電界分布は内部に分布する固定電荷（イオン化した不純物原子）と粒子を用いてポアソン方程式で計算する。流体型シミュレータと同様に、端子電圧を設定すれば、素子の電気的特性を求められる。

粒子の挙動は、散乱とそれにとまうエネルギー変化と進行方向の変化で決定する。基本的には以下に示すボルツマン輸送方程式を解くわけで、散乱前の波数 k の状態から、散乱確率 W により散乱して、 k' へ遷移する。ここで f は分布関数、 F は電界である。

$$\left(\frac{\partial}{\partial t} + \frac{e}{\hbar} F \cdot \frac{\partial}{\partial k} \right) f(k, t) = \int W(k, k') f(k', t) dk' \quad (18)$$

輸送方程式を直接解くことができないので、右辺の衝突項を乱数を用いて決定し、粒子の状態を近似的に求める。衝突の種類として、格子原子の音響と工学フォノン散乱、不純物原子散乱、プラズモン散乱、GaAsの場合は谷間散乱が加わる。各種散乱の確率と散乱方向はエネルギーの関数であたえ、乱数に従って各粒子の挙動を決定する。

個々の粒子の挙動はこのように解くが、移動度や拡散係数などを得るには、数千個以上の粒子を取り扱い、統計的に処理する。このため、計算量が非常に大きくなるが、スーパーコンピュータの導入は比較的容易である。ベクトル化は衝突確率、散乱方向とそれを決める乱数を1次元配列のベクトルとする。粒子の数がベクトル長となるので加速率も大きい。

7. ま と め

半導体デバイスシミュレータの計算方法と、スーパーコンピュータで実行する場合の手法について述べ

た。計算方法は前処理付き共役勾配法系の行列計算を比較検討した。この結果、Decoupled 法ではポアソン方程式から形成される対称行列には ICCG (2, 4) が、電流連続方程式から形成される非対称行列には ILUCGS (2, 4) が優れていた。Coupled 法には BILUCGS (1, 2) が優れていた。これら計算方式はスカラ計算においてもベクトル計算においても最短の時間で計算を行った。

しかし、ベクトル化は計算機の構成に依存性が高く、使用する計算機に合わせてプログラムを最適化する。今後は並列計算機を含めて、計算方法に合わせた計算機の実装も必要であろう。

有限要素法は現状ではあまり使われていないが、将来の超高集積素子のデバイスシミュレータには使用される可能性がある。

参 考 文 献

- 1) 羽根邦夫：半導体シミュレーション技術の動向，電気学会論文誌 C, Vol. 107-C, No. 6, pp. 496-501 (1987).
- 2) Scharfetter, D. L. and Gummel, H. K.: Large-Signal Analysis of Silicon Read Diode Oscillator, IEEE Trans. on Electron Devices, Vol. ED-16, No. 1, pp. 64-77 (1969).
- 3) Hestenes, M. R. and Seinfeld, E.: J. Res. N. B. S. 49, No. 6 (1952).
- 4) Meijerink, J. A. and van der Vorst, H. A.: An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix Is a Symmetric Matrix, Math. Comput., Vol. 31, pp. 148-162 (1977).
- 5) Gustafsson, I.: A Class of First Order Factorization Method, BIT 18, pp. 142-156 (1978).
- 6) Saad, Y.: The Lanczos Biorthogonalization Algorithm and Other Oblique Projection Methods for Solving Large Unsymmetric Systems, SIAM J. Numer. Anal., 49, pp. 458-506 (1982).
- 7) Heiger, C.: Preconditioned Iterative Methods for Nonsymmetric Large Systems, Proc. Int. Conf. Simulation of Semiconductor Device and Processes, pp. 267-285 (1984).
- 8) 佐藤成生, 羽根邦夫, 鈴木登紀男：デバイスシミュレータ用の高収束な行列計算方法，電子通信学会論文誌 C, Vol. J69-C, No. 11 (1986).
- 9) Abe, G., Hane, K. and Suzuki, T.: Device Simulation Code for Vector Processor, NASE-CODE IV Boole Press (1985).

(昭和 62 年 8 月 13 日受付)