二次元複合デバイスシミュレータのための 高速反復行列解法

正 員 中 村 慎 (東 芝)
正 員 中 川 明 夫 (東 芝)

1. はじめに

近年パワー素子は応用分野が広がり市場が拡大している。パワー素子を設計・開発するために計算機によるシミュレーションを利用することができれば、LSIの場合そうであるように効果は大きいと考えられる。しかし LSI のシミュレーションに比べて、パワー素子のシミュレーションは高耐圧・高注入状態を扱う必要があるため収束解が得にくく、そのために実際問題への適用は遅れていた。そこで筆者らは高耐圧・高注入状態を解析できるバイポーラ・MOS 複合デバイスのための収束性の良い汎用二次元シミュレータ TONADDE II (Toshiba Numerical Analysis program for Device Design) を開発してきた⁽¹⁾⁽²⁾。しかしそれには計算時間が、特に大形問題に対して長く掛かるという欠点があった。そこで今回、このシミュレータに組込むことでより高速化がはかれる解法を開発した。

TONADDE II では、結合解法を用いており、また、完全ニュートン法を行うために 9 点差分法を用いている。それによって、ニュートン法の収束が加速されバイアス条件にかかわらず 5 回以内のニュートン反復で解が得られるという特徴がある。しかし、結合解法を使用することで必然的に大形連立一次方程式を解なくてはならない。その解法としては従来直接法（ブロック消去法）が使用されてきたが、直接法は差分格子数が大きくなると計算時間が急激に長くなるので、より高速な大形線形連立方程式解法が必要になりこう配法、反復法の使用が検討されてきた。高速反復解法と

して CG 法 (Conjugate Gradient method) 系統の方法が知られているが結合解法に対して用いられた例はきわめて少ない⁽³⁾。

本論文では大形連立方程式を解くために不完全 LU 分解による前処理を用いた共役こう配法系統の解法である CGS 法 (Conjugate Gradient Squared method) および BCG 法 (BiConjugate Gradient method) を 9 点差分・結合解法用に 3×3 小行列を 1 単位（ブロック）として考えて適用することで、直接法に比べて高速に解が得られることがわかったので報告する。

2. 非線形系解法の概要

一般にデバイスシミュレータでは、以下のようなポアソン式と電流連続式を解いている。

$$\text{div grad } \phi = -\frac{e}{\epsilon} (N_d - N_a + p - n) \dots (1)$$

$$\frac{\partial p}{\partial t} = -\frac{1}{e} \text{div } \mathbf{J}_p + G_p - U_p \dots (2-1)$$

$$\frac{\partial n}{\partial t} = \frac{1}{e} \text{div } \mathbf{J}_n + G_n - U_n \dots (2-2)$$

$$\mathbf{J}_p = -e D_p \text{grad } p - e \mu_p p \text{grad } \phi$$

$$\mathbf{J}_n = e D_n \text{grad } n - e \mu_n n \text{grad } \phi$$

(記号は文末参照)

これらの微分方程式の解法は大別して、方程式を一つずつ解く方法 (decoupled method) と三つの式を同時に解く結合解法 (coupled method) がある (図 1) が、 (p, n, ϕ) が緊密に依存しあいながら計算が進められるので、収束性がよい後者を採用している。

特に 9 点差分法と組み合わせると高注入の場合にもニュートン法の反復回数を 5 回以内に抑えることが可能である。ここで 9 点差分とは、図 2 の点 B で微分方程式を離散化する際、点 A, C, E, H での変数だけでなく D, F, G, I での変数も含めて最大 27 個の微係数

A Rapid Convergence Iterative Matrix solver for 2-D Composite Device Simulator. By Shin Nakamura, Member & Akio Nakagawa, Member (R&D center, Toshiba corporation)
中村 慎: 正員, (株) 東芝総合研究所
中川明夫: 正員, (株) 東芝総合研究所

図 1 結合解法(右)と非結合解法の処理の流れ

Fig. 1. Flow charts for coupled method (right) and decoupled method.

図 2 9点差分のための格子

Fig. 2. Grid system for 9-point discretization.

すべてを、ニュートン反復で使用するヤコビアンで考慮するというのである。例えば、 x 方向の電場は格子点の中間(例えば点 M)で点 A, B, C, G, H, I での6つの ϕ から定義される。

$$E_x(M) = \frac{(\phi_B + \phi_H - \phi_C - \phi_I) \frac{X_{AB}}{X_{BC}}}{2(X_{AB})} * \frac{(\phi_A + \phi_G - \phi_B - \phi_H) \frac{X_{BC}}{X_{AB}}}{* + X_{BC}}$$

図 3 バイポーラモード MOSFET の構造
Fig. 3. Bipolar-mode MOSFET structure.

図 4 バイポーラモード MOSFET の電流
電圧特性(破線)を計算する
際のニュートン反復回数

Fig. 4. Newton iteration number vs. applied drain voltage (solid line) when the model simulates current-voltage characteristics (dashed line) for bipolar-mode MOSFET.

.....(3)

(X_{AB} は点 A, B 間の距離を示す。)

従って移動度が電界に依存するとする定式化を採用すると、移動度の微分には必然的に隣接する9点での変数が含まれる。

実際に9点差分および従来の5点差分を用いて、図3のような600Vバイポーラ形MOSFETの電流電圧特性を計算した場合のニュートン反復回数を、図4に示した。9点差分を用いることで5点差分を用いる場合に較べてニュートン反復の収束が加速され、電流連続の式において5けたの連続性を保つという条件のもとでバイアス条件にかかわらず5回以内で収束している。

3. 線形方程式解法

しかしながら、結合解法を用いると、従来の方程式

図 5 9点差分法により得られるヤコビ行列 (図は 3×3 行列を示す.)

Fig. 5. Jacobian matrix obtained from 9-point discretization method.

をひとつずつ解く非結合解法に較べて3倍の自由度の線形方程式(4)を解く必要がある。図2のように x -方向に N_x 分割, y -方向に N_y 分割されたメッシュに対して離散化によって得られるニュートン反復にあらわれるヤコビアン A は図5のような非対称規則的疎行列で大きさは $3N_xN_y$ と大きくなる。

$$Ax = b \dots\dots\dots (4)$$

ここに図は 3×3 小行列を現しており, 行列 A の 1 行に 27 個の非零要素がある。 x は未知の p, n, ϕ の並びのベクトルである。

従来は高注入・高耐圧の場合この大形連立一次方程式の解法としては, 直接法(ガウスの消去法, ブロック消去法)がよく使用されてきた。直接法は確定回数 of 演算で確実に解が求まる, スーパーコンピュータを使用する際ベクトル化されやすく, スーパーコンピュータの性能をいかしやすいなどの長所はあるものの, 解くべき問題が大きくなるにつれて, 計算時間・必要な記憶容量が急激に増大するので結合解法を用いる場合のような大形問題では実用的でない。また, SIP (Strongly Implicit Procedure), SLOR (Successive Line Over Relaxation method) は今回のような問題においては良い収束性を示さない⁽⁴⁾。

ところで近年種々の分野で使用されている方法に共役こう配法系統の解法があり, それらを適当な前処理と組合わせた解法は ICCG (Incomplete Cholesky Conjugate Gradient method), PCG (Preconditioned CG) などと呼ばれ広く使用されてきている⁽⁵⁾⁽⁶⁾。特にいまの場合は共役こう配法系統の解法の中でも非対称行列を係数とする連立方程式の解法であ

る BCG 法, CGS 法, CR 法 (Conjugate Residual method) をブロック不完全 LU 分解 (3×3 小行列を単位として考えて不完全 LU 分解を行う) による前処理とともに 9 点差分・結合解法に適用した。

共役こう配法を適用する手続きは次のようになる。連立方程式(4)を解くためにまず, 行ごと, 列ごとのスケーリングを行う。列ごとのスケーリングは, 今のように電子密度・正孔密度のような 10 の 15 乗から 20 乗のオーダーの量と電位のようなオーダーが高々 10 の 3 乗の量を混在させて解く場合には必要な手続きであり省くと収束性が悪くなる。スケーリングを行った後の係数行列をあらためて A とおきブロック不完全 LU 分解を行う。つまり SIP の場合と同様に図6のような非零要素を持つ下三角行列 L , 上三角行列 U を仮定し, 積 $L \cdot U$ を計算し A と比較して L, U の要素を定める。

$$A = L \cdot U + R \quad R \text{ は無視する} \dots\dots\dots (5)$$

図6で b, c, d などは 3×3 の小行列の連なりであり, 次のように計算される。この演算においてはすべて 3×3 行列がひとつの要素であるかのように考える。ここで b_i は 3×3 の小行列の連なり b の i 番目の 3×3 小行列の連なりを示す。また, $m = N_x$ とする。

$$b_i = E_i, \quad v_i = F_i - b_i e_{i-m}$$

$$x_i = -v_i e_{i-m+1}$$

$$c_i = A_i - b_i w_{i-m} - v_i y_{i-m+1}$$

図 6 不完全 LU 分解のために仮定した L, U 行列 (1つの線は 3×3 小行列の連なりを示す.)

Fig. 6 Assumed L, U matrix for incomplete LU decomposition.

$$\begin{aligned}
d_i &= B_i - c_i e_{i-1} - b_i f_{i-m} - v_i w_{i-m+1} \\
&\quad - x_i y_{i-m+1} \\
e_i &= d_i^{-1} (C_i - v_i f_{i-m} - v_i w_{i-m+1} \\
y_i &= d_i^{-1} (-c_i w_{i-1}) \\
w_i &= d_i^{-1} (G_i - c_i f_{i-1}) \\
f_i &= d_i^{-1} H_i \dots\dots\dots (6)
\end{aligned}$$

次に BCG/CGS/CR 法を次のような問題に対して適用する。

$$(LU)^{-1}Ax = (LU)^{-1}b \dots\dots\dots (7)$$

$$A(LU)^{-1}(LU)x = b \dots\dots\dots (8)$$

$$L^{-1}A U^{-1}Ux = L^{-1}b \dots\dots\dots (9)$$

この論文では図6の L, U 行列を使用した BCG 法を BCG (1, 3) と呼ぶことにする。図6で x, y を無視した場合は BCG (1, 2), 更に v, w を無視した場合は BCG (1, 1) と呼ぶことにする。CGS 法の場合も同様にする。なお CR 法の場合、6図の L, U を使用し、直交法を保持するために n 本のベクトルを用いる方法を CR (n) と呼ぶことにする。

直接法と共役こう配法の計算時間・記憶容量について考察する。対象とする領域の x, y 方向差分格子分割数を N_x, N_y とする。このとき直接法と共役こう配法の必要とする計算時間および記憶容量のオーダーを N_x, N_y の関数として表1に示した。ただし、 $N_x > N_y$ の場合は方程式の順序、変数の順序の入替えによっていつでも $N_x < N_y$ にすることができる。ここで α, β などは計算機・プログラムに依存する定数である。直接法の場合は大きさ $3N_x$ の小行列の逆行列を求めるためには計算量の理論より $(3N_x)$ の3乗に比例する時間がかかるが、それを N_y 回求める必要がありこれが計算時間のほとんどを占めることから見積もった。これらは数値的な演算のみならず、分岐・判定なども含めて考えている⁽⁷⁾。また、共役こう配法の場合、大きさ $3N_x N_y$ の行列に対するスケーリング・不完全 LU 分解を行う前処理部分と、大きさ $3N_x N_y$ の図5のような疎行列とベクトルの積の計算が主となる反復部分が計算時間の主要な部分を占めること

表1 $N_x \times N_y$ 差分格子に対する必要な記憶容量・計算時間のオーダーの見積り (N_x, N_y が大きい場合)

Table 1. Order estimation of storage space and computation time.

	計 算 時 間	記 憶 容 量
直接法 (ブロック消去法)	$\alpha N_x^3 N_y$	$9 N_x^2 N_y$
BCG (1,3)	$(\beta + \gamma m) N_x N_y$	$\delta N_x N_y$

m : 反復回数, $\alpha, \beta, \gamma, \delta$ は定数

から見積もることができる。

これらの見積り式は N_x, N_y が十分大きな場合に一般的にそのオーダーを示すものであり N_x, N_y が小さい場合には当てはまらない。しかし、実際に使用される格子数である N_x, N_y が 20-100 程度の値の場合でもこれらの式を使用することが可能である。その場合 α, β などは N_x, N_y などの関数となるが、実用上許容できる範囲内でそれらを定数とみなしてよいことを示すことができる。

例えば BCG (1, 3) の場合に自由度 ($= 3N_x N_y$) をかえて実際に計算時間を測定し表1の式から逆に β, γ を求めてプロットしてみると図7のようになる。 N_x, N_y が小さいところでは誤差が大きいが計算時間のオーダーを見積もるには十分である。直接法についても同様である。

実際にわれわれの使用した計算機 CRAY-XMP では計算速度に関する指標 α, β などは次のような値をとる。

$$\alpha = 6E-7, \beta = 5.6E-5, \gamma = 1.4E-5 \dots (10)$$

従って、表1は計算時間・記憶容量でも共役こう配法が直接法に比べて N_x, N_y が大きい場合に有利であることを示している。また、ここから議論のための仮定として 40 回で収束するとすれば、 $N_x > 30$ ならば直接法より共役こう配法のほうが効率が良いことがわかる。また、この値は 100×100 の差分格子 (30,000 自由度) を用いた場合、連立方程式が、例えば BCG 法で CG 反復が 40 回で収束するとすれば、表1より

$$\begin{aligned}
&(\beta + \gamma m) N_x N_y \\
&= (5.6E-5 + 1.4E-5 \times 40) \times 100 \times 100 = 6.16
\end{aligned}$$

よって約6秒で解け、従って、ニュートン反復5回で収束するとすれば1バイアスあたり30秒程度で計算できることになる。これは、同様に直接法で連立方程式が60秒、従って、1バイアスあたり300秒かかる

図7 β, γ の値と自由度の関係

Fig. 7. β, γ v.s. degree of freedom.

表 2 実用的な問題の場合の計算時間の例

Table 2. Examples of computation time for practical problems.

格子数 ($a \times b$) (自由度)	23×81 (5589)	27×106 (8586)	70×61 (12810)
直接法 (逆行列計算)	0.98 秒 (0.77 秒)	1.8 秒 (1.4 秒)	13 秒 (12 秒)
BILUBCG (反復回数)	1.2 秒 (44 回)	3.9 秒 (83 回)	2.5 秒 (45 回)

こととくらべると 10 倍高速である。なお 40 回の反復で収束するというのは比較のための仮定である。経験的にも 40 回程度で収束する場合が多いが問題によっては 100 回程度の反復を必要とする場合もある。しかし、その場合でも直接法が有利な N_x の範囲が少し広がるだけで、 N_x , N_y が大きい場合に直接法より共役こう配法のほうが効率が良いという結論に変わりはない。

表 2 はいくつかの現実の応用で現れる異なるデバイスにたいして直接法と BCG (1, 3) との計算時間を比較したものであるが、以上の見積りが正当なものであることを示している。

4. ベクトル化

こうした計算をスーパーコンピュータで実行するためにはベクトル化がどの程度なされるかが計算時間短縮のために重要な因子となる。共役こう配法 BCG (1, 3) などを使用する場合には上記アルゴリズムをコーディングした場合に最内側の 3×3 行列の演算はインライン展開 (DO ループを一部分展開) することで最内側ループのベクトル長を長くすることができる。また、不完全 LU 分解を使用した前進・後退代入 (例えば, $q = (LU)^{-1}r$) などではそのままではベクトル化できないので超平面法などの手法が考えられてきている⁽⁸⁾。我々も超平面法を TONADDE II に適用した。すなわち、演算順序は例えば $U^{-1}r$ の計算の場合図 8 にあるような依存関係があるので図 9 にあるよ

●での変数が計算できるためには、○での変数が計算されていなくてはならない。

図 8 計算の依存関係

Fig. 8. Dependence of computation.

1つの線上の格子点での変数は同時に計算できる。

図 9 超平面法

Fig. 9. Hyper plane method.

図 10 実用問題での ILUBCG/CGS/CR 法の収束特性

Fig. 10. Convergence characteristics for ILUBCG/CGS/CR method in practical application.

うな順序にすることでベクトル化を可能にする。しかし、このようにするとベクトルの長さは平均してほ

$$\frac{N_x N_y}{N_x + 3 N_y} \dots\dots\dots (11)$$

とそれほど長くない。それでも実際にテストしてみた結果では、表 2 にある程度の格子系にたいしてスカラモードと比較して高々 2~3 倍の高速化はなされる。つまり (10) の値は $1/2 \sim 1/3$ になる。このとき第 3 章で示した直接法、共役こう配法の効率に関する議論はそのまま成り立つ。

5. 応用例

このシミュレータを図 3 に示すような実際のバイポーラ形 MOSFET の解析に適用した場合の定常解析での BCG/CR/CGS 法の収束特性の一例を図 10 に示す。差分格子は 70×61 である。この他のいくつか

図 11 TONADDE II を使用したときの
電流電圧特性の例

Fig. 11. An example of current-voltage curve
computed by TONADDE II.

の場合に適用してみた結果では BCG/CGS 法はいつでも収束したが CR (CR (6), CR (12)) 法ではこの図のように収束しない場合があった。ただし、収束する場合には 1 回あたりの反復にかかる時間が CR 法では BCG/CGS 法の 6 割程度で済むので収束する場合には高速な場合もあった。直交するベクトルをさらに増やすことも可能であるが、その場合、必要な記憶容量は増加するが収束性はほとんど改善されなかった。

BCG (1, 3) と BCG (1, 2) では BCG (1, 3) が BCG (1, 2) よりも収束性がよいが BCG (1, 3) はより多くのベクトルを使用しているため前処理・一回の反復にかかる時間が長いので全体としての計算時間の減少は反復回数の減少ほどにはならない。BCG 法と CGS 法を比較すると CGS 法が BCG 法より多くの場合速く収束した。図 11 は TONADDE II を使用して描いた電流・電圧特性の例である。

6. 結 論

バイポーラ MOSFET の解析のための非線形反復

解法として結合解法と完全ニュートン法のための 9 点差分の組合せによる方法が、高バイアス条件下において収束性がよく、そのための大形線形方程式の解法としてブロック不完全 LU 分解による前処理をもちいた共役勾配法 (BCG/CGS) が直接法に比べて効率がよい解法であることを示した。

(昭和 62 年 2 月 2 日受付, 同 62 年 10 月 13 日再受付)

文 献

- (1) A. Nakagawa, D.H. Navon: "A Time-and Temperature-dependent 2D Simulation of the GTO Thyristor Turn-Off Process" *IEEE Trans. Electron Devices*, ED-31, pp. 1156 (1984)
中川・中村: 「 p, n, ψ を結合させて解く高速反復解法を用いた複合素子モデル TONADDE II」 昭 62 電子通信学会大会(春)予稿集
- (2) 中川: 「パワーデバイスと複合デバイスのシミュレーション」 *電学論 C*, 107, No. 6 (昭 62-6)
- (3) 佐藤・羽根: 「デバイスシミュレータのための高収束な行列計算法」 *電子信学論誌 J69-C*, No. 11 (昭 61-11)
- (4) A. Yoshii, H. Kitazawa, M. Tomizawa, S. Horiguchi & T. Sudo: "Three dimensional analysis of semiconductor devices" *IEEE Trans. Electron Devices* ED-29, 2, 184 (1982)
- (5) Y. Saad: "The Lanczos biorthogonalization algorithm and other oblique Projection methods for solving large unsymmetric systems" *SIAM J. Numer. Anal.* V. 5, pp. 530 (1968)
- (6) 野寺: 「一般 CG 法の誤差の評価式について」 *情報処理学会 27 回全国大会予稿集* pp. 1273
- (7) 唐木: 「スーパーコンピュータにおける演算量の見積り」 *大型行列計算シンポジウム* p. 36 (昭 60-9)
- (8) 日本物理学会編: 「スーパーコンピュータ」 (昭 60) 培風館
- (9) 倉田: 「バイポーラトランジスタの動作理論」 (昭 55) 近代科学社

付 録

(記 号 説 明)

ψ : 電位, e : 電子電荷絶対値, ε : 誘電率, N_A , N_D : ドナー, アクセプタ濃度, p : 正孔密度, n : 電子密度, J_p : 正孔電流密度ベクトル, J_n : 電子電流密度ベクトル, G_p , G_n : 正孔, 電子単位時間あたり発生率, U_p , U_n : 正孔, 電子単位時間あたり再結合率, D_p , D_n : 正孔, 電子の拡散定数, μ_p , μ_n : 正孔, 電子の移動度