ハイパフォーマンスコンピューティング　60－5
（1996．3．4）

強制疎化 LU 分解法による回路シミュレーション

須田 礼仁, 小柳 義夫
東京大学 理学部 情報科学科

半導体技術の進歩にともない、大規模回路シミュレーションの重要性はいよいよ高まっている。特に連立一次方程式の求解は回路規模に対して線形以上の計算量を消費し、並列化もしにくいため大規模回路シミュレーションの最大の課題の一つである。本研究では不完全 LU 分解を時間刻み毎に修正し、反復改良で解を求める強制疎化 LU 分解を提案する。この方法は前処理つき緩和法、直接法、不完全 LU 分解法などを包含する一般化された方法である。この方法は直接法で解ける全ての問題を解くことが可能であり、高い精度の必要ない場合には少ない計算量で済む反復法的な性質もあわせもっている。動的でない部分回路に関する計算を行なわないので回路のレイテンシを有効に利用して計算量を減らすことができた。また直接法に比べて計算のクリティカルパスが短くなるので並列処理にも有効であると期待される。

The Ensparsed LU Decomposition Method for Circuit Transient Analysis

Reiji Suda and Yoshio Oyanagi
Department of Information Science, Faculty of Science, the University of Tokyo

Along with the progress of the semiconductor technology, large scale circuit simulation gains in its importance. A largest problem of large scale circuit simulation is the linear solution, because its computational complexity is higher than linear to the size of the circuit size, and efficient parallel implementation of the linear solver is not an easy task. This paper proposes the Ensparsed LU (ELU) decomposition method, which modifies an incomplete LU decomposition at each time step, and obtaines the solution by relaxation. The ELU method encompasses the preconditioned relaxation methods, the direct method, and the imcomplete LU decomposition methods. It can solve any equation which a direct method solves, requires lesser computational cost when the precision is lower, and exploits the latency of circuits by bypassing the computation about the latent subcircuits. It will be effective for parallel processing, because the critical path of the computation becomes shorter than the direct method.

§1 はじめに

半導体および回路設計において大規模回路シミュレーションの重要性は高まっている。大規模回路シミュレーションのために必要な技術はさまざまであるが、その中の一つに連立一次方程式の求解がある。回路シミュレーションにおける連立一次方程式を解くためには、広く直接法が用いられている。しかしこの計算量は回路規模に対して線形以上で増大するため、大規模回路では計算量を抑える工夫が重要である。また大規模シミュレーションに必要な計算性能を並列化で工面する必要も出てくるであろうが、直接法は並列性が低く回路シミュレーションの並列化における最大の問題である。

これに対して反復解法を回路解析に用いる提案がいくつもなされてきた。一般に反復法は回路の疎行列性を活かすことが容易であるため、回路規模に伴う計算量の増加が直接法に比べて緩やかである。また効率的な並列化も直接法に比べ容易であるという利点がある。これに対し反復法の最大の問題点は、回路によっては収束が遅かったり、まったく収束しなかったりすることである。このような欠点を克服するためには直接法をなんからの形で取り込むことが必要であり、提案されている反復法の前処理はいずれも直接法を利用している。

回路解析に対して提案された主な反復解法は以下の通りである。共役勾配法の系統では反復あたりの計算量の少ない CR 法 [12] と収束の早い CGS 法 [1, 7] が提案されている。前処理としては ILU(ℓ) [1, 7] のほか、以前の時間刻みにおける LU 分解 [12] が提案されている。緩和法の系統では単純な Gauss-Seidel 法を常微分方程式のレベルに適用した波形緩和法 [5] が広く注目されている。この方法は適用できる回路の範囲が非常に狭いため、改良した解法がいくつか提案されている [4, 8]。緩和法に前処理を行なう前処理つき緩和法 [10] も提案されている。また、近似的な LU 分解を用いて緩和法で解を求める方法もいくつか提案されている。近似的 LU 分解としては不完全 LU 分解 [2, 6, 3] や LU 分解の部分修正 [9] がある。

本論文では回路過渡解析のための新しい連立一次方程式の解法として強制疎化 LU 分解法を提案する。この方法は上述の近似的 LU 分解による緩和法を一般化したものであり、高い安定性と低い計算量を両立できる方法である。

§2 強制疎化 LU 分解法

§2.1 不完全 LU 分解による緩和法 (ILU 緩和法)

連立一次方程式を行列とベクトルの形で

$$Ax = b \tag{2.1}$$

と表す。不完全 LU 分解は A の近似的な LU 分解で、その誤差 E は

$$LU + E = A \tag{2.2}$$

によって定義される。この E が十分小さい時、緩和法の反復

$$x^{(i+1)} = x^{(i)} + U^{-1}L^{-1}r^{(i)} \tag{2.3}$$

は $x = A^{-1}b$ に収束する。ここで

$$\begin{aligned} r^{(i)} &= b - Ax^{(i)} \\ &= E(x^{(i)} - x^{(i-1)}) \end{aligned} \tag{2.4}$$

は残差である。緩和法の反復は

$$x^{(i+1)} = U^{-1}L^{-1}(b - Ex^{(i)}) \tag{2.5}$$

と書くことも出来る。二つの緩和法の反復式は計算量は同じであるが、後者の方が広く用いられて来た。しかし前者の方が残差が直接求まることや、ここで提案する強制疎化がやりやすいことなど有利な点も多い。

表 2.1. ILU 分解における強制疎化

強制疎化	種類	精度の向上の方法
(1) 行列 A	行列	反復
(2) LU 分解の Fill-in	行列	反復
(3) 行列の時間差 $A_t - A_{t-1}$	行列	反復
(4) 残差 r	ベクトル	弱い強制疎化
(5) 誤差行列 E	行列	弱い強制疎化
(6) 修正 $x^{(i)} - x^{(i-1)}$	ベクトル	弱い強制疎化

§2.2 ILU 緩和法における強制疎化

強制疎化とは、計算の対象であるデータのうち十分小さいものを無視して計算を省略してしまうことである。回路シミュレーションにおける bypassing や、固有値解法としての Jacobi 法の threshoulding も一種の強制疎化であると考えられる。連立一次方程式は行列とベクトルで定義されているので、強制疎化は行列やベクトルの要素に適用されることになる。さらに、ILU 緩和法を用いて連立一次方程式を解くこと、および係数行列 A が時間的に変動するすることを考慮に入れると 表 2.1 にあるような 6 点について強制疎化が可能であることが分かる。

これらのうち (1) から (3) までは LU 分解に関係しており、(4) から (6) までは反復改良に関係している。誤差行列 E と修正 $x^{(i)} - x^{(i-1)}$ の強制疎化は残差 $r = E(x^{(i)} - x^{(i-1)})$ の計算の際の強制疎化である。残差は解の精度の評価に用いられるため、このふたつの精度を低くすることは危険であり、要求されている精度が高い場合には、最後の一回はまじめに $r = b - Ax$ を計算する必要がある。

残差 r の強制疎化は緩和法の反復 LU 分解の代入 $x^{(i+1)} = x^{(i)} + U^{-1}L^{-1}r^{(i)}$ の際に用いる。この場合代入は Prog. 2.1 のように列単位に実行する必要がある。

```
for (i = 1, n)
    y_i = r_i / l_ii
    if (y_i cannot be ignored)
        for (j = i + 1, n)
            r_j = r_j - y_i * l_ij

for (i = n, 1)
    x_i = y_i / u_ii
    if (x_i cannot be ignored)
        for (j = 1, i - 1)
            y_j = y_j - x_i * u_ij
```

Prog. 2.1. 強制疎化を伴う代入

行列 A [2, 6] および fill-in [3] に対する強制疎化は他の ILU 緩和法の研究で取り扱われてきたが、同時に適用した研究はない。いずれの研究でも反復回数がかなりかかっているのでそれ以上近似を加える気になれなかったのであろうか。これらの論文で指摘されているように、行列に対する強制疎化は、回路シミュレーションではかなり有効である。また、枢軸選択をしながら LU 分解を進めてゆくと、行列要素は小さい値を取るようになる傾向があり、このため強制疎化の対象になる fill-in 要素はかなりの数になる。この方法は強制疎化によって非零要素が減るために、それによって生じるはずであった fill-in が生じなくなり、生じた fill-in の強制疎化と共に二重に fill-in を抑制することになるので、LU 分解の計算量の増大を有効に防ぐことができる。

行列の時間差 $A_t - A_{t-1}$ に対する強制疎化を行なうためには LU 分解の修正のアルゴリズムが必要になる。その最も簡単なものは同じ LU 分解を繰り返し使い、使えなくなったら最初から計算し直すもの [12] である。行列を部分回路構造に基づいた再帰的縁つき対角ブロック行列に仮定してブロック単位で修正をする方法が [9] で提案されている。本論文では次節で説明するような LU 分解の修正の方法を用いている。

1	**for** $(i = 1, n)$
2	**for** $(j = 1, n)$
3	$a_{ij} = a_{ij} - \sum_k a_{ik} a_{kk}^{-1} a_{kj}$

Prog. 3.1. LDU 分解のアルゴリズム

§3 LU 分解の修正アルゴリズム

LU 分解の修正のアルゴリズムはいくつか知られており、ランク 1 の修正アルゴリズムは有名である。しかし回路シミュレーションでは修正のランクが 1 という期待はできない。ここではもっと単純に LU 分解を部分的に計算し直す方法を用いる。その基礎になるアルゴリズムは Prog. 3.1 に示す LDU 分解である。

今、行列 A が修正されて A' になったとし、その要素を a'_{ij} とする。この記号は LDU 分解で上書きされたデータにもそのまま用いることにする。すると、LDU 分解の修正 $\delta_{ij} = a'_{ij} - a_{ij}$ は次のように書くことが出来る。

$$\delta_{ij} = \delta_{ij} - \sum_k (a'_{ik} a'^{-1}_{kk} a'_{kj} - a_{ik} a_{kk}^{-1} a_{kj}) \tag{3.1}$$

ここで和の各項は δ を使って

$$a'_{ik} a'^{-1}_{kk} a'_{kj} - a_{ik} a_{kk}^{-1} a_{kj} = \delta_{ik} a_{kk}^{-1} a_{kj} + a'_{ik} \tilde{\delta}_{kk} a_{kj} + a'_{ik} a'^{-1}_{kk} \delta_{kj} \tag{3.2}$$

のように書くことが出来る。ただし $\tilde{\delta}_{kk} = a'^{-1}_{kk} - a_{kk}^{-1}$ である。以上の式にしたがって LDU 分解を修正すれば、$\delta_{ik}, \delta_{kk}, \delta_{kj}$ のどれかが非零となる部分についてのみ計算すればよいことになる。

このような LU 分解の修正は強制疎化を伴ってこそ有効に活用できる。一つの要素の修正が LU 分解の依存関係にしたがって行列の右下に向かって連鎖的に修正を引き起こすことになるが、この連鎖の広がりは強制疎化によってかなり狭く抑えることが出来る。

しかし LU 分解を修正で用いることによっていくつか問題も生じる。一つは枢軸選択の順序が変えられないことである。枢軸選択の順序は計算精度や疎行列性に直接影響するので、場合によっては重大な問題を引き起こす可能性もある。もう一つは時間刻みが変更できないことである。もちろん時間刻みの変更にともなう行列の変化を LU 分解の修正の形で扱うことは出来るが、これはかなり非効率的である。いずれにせよ、必要な場合には最初から LU 分解をしなおさなければならない。

§4 Lazy ELU 法とその性能

強制疎化 LU 分解法は表 2.1 にあるような強制疎化を伴った ILU 緩和法として定義される。ここではそれを少し単純化し、緩和法 (2.3) を反復せずに一度だけ用いる lazy ELU 法を提案し、実装した。この場合近似解は

$$x = U^{-1} L^{-1} b \tag{4.1}$$

という形で求められる。強制疎化としては残差の強制疎化は Prog. 2.1 のような列型代入が実装できなかったので、LU 分解の強制疎化のみを実装した。LU 分解の修正は閾値 θ を用いて

$$|\delta_{ij}| \geq \theta \tag{4.2}$$

を満たす部分を修正している。さらに修正の結果の α_{ij} がθ よりも小さくなった場合は $\delta_{ij} = a_{ij}, a_{ij} = 0$ として零要素にする。閾値 θ は、Newton 法の現在の残差と次の残差の予測値との比に安全係数を掛けて用いた。行列にスケーリングを行なっているのでこのような単純な方法でもかなりうまくゆくようである。実際にはNewton 法の反復回数が増大していないので、かなり厳しいの閾値になっていることがわかる。但し θ がある値 θ_0 よりも小さくなる時は$\theta = \theta_0$ とした。閾値の下限 θ_0 が大きい場合は Newton 法の反復回数が増大してしまうが、θ_0 が小さ過ぎると行列の非零要素が増大してかえって遅くなってしまう。これは一旦非零要素に扱われた要素は θ が大きくなっても修正されるまで零要素にならないことが災いしている。以下の実験では表から分かるように θ_0 を Newton 法の反復回数が 1–2% しか増大しないように選んだ。なお、θ_0 の値は各回路に共通であるが、他の種類の回路 (TTL, MOS, アナログ回路など) でも同じ値で同様の性質が得られるかどうかは今のところ分かっていない。

表 4.1. Lazy ELU 法による回路シミュレーションの所要時間

解法		half	full	2bit	4bit	8bit	16bit
直接法	Newton iter	2.65	2.84	3.07	3.24	3.44	3.59
	linear solver	0.4	1.0	1.5	5.8	23.7	123.0
	matrix setup	0.1	0.3	0.4	2.2	7.5	41.4
	decomposition	0.2	0.5	0.7	2.3	9.2	49.3
	substitution	0.1	0.2	0.3	1.1	6.8	31.7
Lazy ELU 法	Newton iter	2.65	2.84	3.09	3.26	3.46	3.60
	linear solver	0.4	0.7	0.9	3.3	14.4	63.6
	matrix setup	0.1	0.2	0.3	1.3	5.1	27.0
	decomposition	0.1	0.2	0.2	1.0	3.8	13.8
	substitution	0.1	0.1	0.3	0.9	5.3	22.4

表 4.1 は lazy ELU 法と直接法による回路シミュレーションの所要時間の比較である。シミュレーションしたのは桁上げ先見回路つきの ECL 加算器である。上半分は直接法の結果、下半分は lazy ELU 法によるもので、それぞれ Newton 法の反復回数、連立一次方程式の求解の時間、および連立一次方程式において行列の作成、LU 分解、代入の各部分の所要時間を示してある。なお、枢軸選択には Markowitz-Tewarson 法を用いた。

ELU 法は直接法に比べ、小さい回路で 1.5 倍、大きな回路では 2 倍以上の速度があることがわかる。内訳から分かるように、ほとんどの速度向上は行列分解の部分の寄与であって、大きな回路では直接法の 1/4 の時間しかかかっていない。今回実験した中で最大の 16-bit 加算器でもわずか 1586 ノードであるから、広く行なわれている数万ノード規模の回路シミュレーションでは、相当な速度向上が見込まれることになる。他の部分も直接法よりは早いが、分解部分とは異なり、速度の比は回路の大きさには依存していない。しかし代入部分の速度は列型の実装を行なって残差ベクトルに対する強制疎化を実現すれば向上することが期待できる。回路が大きくなるほど ELU 法が有利になるのは、latent な部分回路の割合が増大するためであると考えられる。LU 分解の修正のおかげで回路の latency を有効に利用できるからである。残差の強制疎化は実装していないが、やはり latent な部分回路については残差が小さいことが期待できるので、回路の規模にしたがって効果が高くなることが期待できる。また行列作成の時間も bypassing を利用して latency を活用することが出来ると考えられる。

§5 さいごに

最近不完全 LU 分解と緩和法を組み合わせる連立一次方程式の解法が回路シミュレーションのために提案されてきている。本論文ではこれらの方法を一般化したアルゴリズムと言える強制疎化 LU 分解法を提案し、性能を評価した。強制疎化 LU 分解法は直接法の 1.5 倍から 2 倍速く、回路規模に従って有効性が高まることが分かった。これは回路の latency を有効に利用できるからと考えられる。今後、残差ベクトルの強制疎化、bypassing の実装、LU 分解の修正アルゴリズムの改良、また並列実装の研究を進める予定である。

参考文献

[1] R. Burch, P. Cox, et al., "PGS and PLUCGS — Two New Matrix Solution Techniques for General Circuit Simulation," *Proc. ICCAD*, 1989, pp. 408–411.

[2] R. Burch, P. Yang, P. Cox, and K. Mayaram, "A New Matrix Solution Technique for General Circuit Simulation," *IEEE Trans. CAD*, Vol. 12, No. 2, pp. 225–241, Feb., 1993.

[3] K. M. Eickhoff and W. L. Engl, "Levelized Incomplete LU Factorization and Its Application to Large-Scale Circuit Simulation," *IEEE Trans. CAD*, Vol. 14, No. 6, pp. 720–727, Jun., 1995.

[4] W. Fang, M. E. Mokari, and D. Smart, "Robust VLSI Circuit Simulation Techniques Based on Overlapped Waveform Relaxation," *IEEE Trans. CAD*, vol. 14, no. 4, Apr. 1995, pp. 510–518.

[5] E. Lelarasmee, A. Ruheli, and A. L. Sangiovanni-Vincentelli, "The waveform relaxation method for the time-domain analysis of large scale integrated circuit," *IEEE Trans. CAD*, Vol. 1, No. 3, Aug. 1982, pp. 131–145.

[6] A. Lumsdaine, J. White, D. Webber, and A. Sangiovanni-Vincentelli, "A Band Relaxation Algorithm for Reliable and Parallelizable Circuit Simulation," *Proc. ICCAD*, pp. 308–311, 1988.

[7] K. Mayaram, P. Cox, et al., "A Parallel Block-Diagonal Preconditioned Conjugate-Gradient Solution Algorithm for Circuit and Device Simulations," *Proc. ICCAD*, pp. 446–449, 1990.

[8] P. Saviz and O. Wing, "Circuit Simulation by Hierarchical Waveform Relaxaion," *IEEE Trans. CAD*, vol. 12, no. 6, 1993, pp. 845–860.

[9] R. Suda, *New Iterative Linear Solvers for Circuit Simulation*, master thesis, Univ. Tokyo, Jan. 1993.

[10] R. Suda and Y. Oyanagi, "Implementation of sparta, a Highly Parallel Circuit Simulator by the Preconditioned Jacobi Method, on a Distributed Memory Machine," *Proc. ICS*, pp. 209–217, 1995.

[11] J. White and A. Sangiovanni-Vincentelli, "RELAX2: A new waveform relaxation approach for the analysis of LSI MOS circuits," *Proc. 1983 Int. Symp. Circ. Syst.*, May 1983.

[12] F. Yamamoto, Y. Umetani, S. Takahashi, "Applicability of Conjugate Residue with Complete LU decomposition for Large-Scale Circuit simulation," *Trans. IPSJ*, Vol. 27, No. 8, Aug. 1986, pp. 774–782 (in Japanese).