An Efficiently Preconditioned GMRES Method for Fast Parasitic-Sensitive Deep-Submicron VLSI Circuit Simulation*

Zhao Li and C.-J. Richard Shi
Department of Electrical Engineering, University of Washington
Seattle, WA 98195
{lz2000, cjshi}@ee.washington.edu

Abstract: *We propose an efficiently preconditioned generalized minimal residual (GMRES) method for fast SPICE-accurate transient simulation of parasitic-sensitive deep-submicron VLSI circuits. First, when time step-sizes vary within a predefined range, the preconditioned GMRES method is applied to solve circuit matrix equations rather than LU factorization. The preconditioner we use comes directly from the previously factorized L and U matrices. Second, to keep using the same preconditioner during nonlinear iteration, the successive variable chord method is applied as an alternative to the Newton-Raphson method. An improved piecewise weakly nonlinear definition of MOSFETs is adopted and the low-rank update technique is implemented to refresh the preconditioner efficiently. With these techniques, the number of required LU factorizations during transient simulation is reduced dramatically. Experimental results on power/ground networks have demonstrated that the proposed method yields SPICE-like accuracy with an about 18X overall CPU time speedup over SPICE3 for circuits with tens of thousands elements.*

1. Introduction

In modern deep-submicron VLSI circuit design, parasitic effects are no longer ignorable with the higher operation frequency, lower supply voltage and smaller device feature size [10]. Accurate post-layout verification requires full-chip simulation of large-scale circuits together with massive extracted parasitic elements, which come from substrate, power/ground networks, interconnects, etc. For such kind of circuits, the per-iteration cost with SPICE [9] during transient simulation is dominated by costly LU factorizations. New approaches are required for fast-yet-accurate simulation of parasitic-sensitive circuits. For example, several efficient methods [2][4][5][12] have been proposed to speedup the simulation of power/ground networks. These methods, although orders of magnitude faster than SPICE, are mainly tailored for *purely linear* circuits and have difficulties with nonlinear circuits incorporated for full-chip simulation. Recently, [7] proposed to couple nonlinear circuits together with power/ground networks based on the Gauss-Seidel style relaxation [8]. However, when nonlinear and linear circuits are strongly coupled, the number of nonlinear iterations could be very large.

A key approach to improving the efficiency of SPICE-accurate simulation of large-scale parasitic-sensitive circuits is to reduce the number of LU factorizations during time-domain simulation. In [1], a fixed time step-size is assumed and the successive chord method [8] is applied to linearize nonlinear devices. Thus, only one LU factorization is required during the whole transient simulation. However, the application of this method is limited since excessive nonlinear iterations might be required with only a single chord defined for the entire operating region of nonlinear devices. Further, the assumption of a fixed time step-size is not adequate for circuits with widely distributed time constants.

Recently, we proposed SILCA [6] for VLSI circuits containing strong parasitic coupling effects. Two linear-centric ideas were used to help keep a circuit matrix constant during transient simulation with variable time step-sizes: 1) Semi-implicit iterative integration scheme to keep equivalent conductance of capacitor/ inductor companion models constant for a large time interval; 2) Successive variable chord method to keep linearized conductance of nonlinear devices constant for a large voltage/current range. With these two ideas, the number of LU factorizations can be reduced dramatically. However, as pointed out in [6], the absolute stability region of the iterative integration corrector is related to the number of iterations. Consequently, the stability of the iterative trapezoid formula could become worse than that of the standard trapezoid formula in practice. Therefore, heuristic measures have been incorporated in SILCA to ensure the A-stability at the cost of more iterations and/or more LU factorizations.

The proposed preconditioned GMRES method [11] in this paper borrows the ideas in SILCA to reduce the number of LU factorizations during variable time step-size transient simulation. Instead of using semi-implicit integration predictor and iterative integration corrector in

* This research was supported by DARPA NeoCAD Program under Grant No. N66001-01-8920, an NSF CAREER Award under Grant No. 9985507, and SRC/NSF's Joint Mixed-Signal Initiative under Contract CCR0120371.

1530-1591/05 $20.00 © 2005 IEEE

SILCA, standard integration formulas are applied here. Thus, the stability problem encountered in SILCA is naturally avoided. Furthermore, the number of nonlinear iterations with the preconditioned GMRES method is smaller than that with SILCA and could be comparable to that with SPICE.

It is well known that the key to fast convergence of the GMRES method is to design an efficient preconditioner [11]. A good preconditioner should be as close to the inverse of a circuit matrix as possible and easy to derive. In this paper, we reuse the previously factorized L and U matrices as the preconditioner in transient simulation. The details are described as below,

1) When time step-sizes h_n vary within a predefined range of the basis time step-size h, the preconditioned GMRES method rather than LU factorizations is applied to solve circuit matrix equations. The preconditioner comes directly from the previously factorized L and U matrices base on the basis time step-size h.
2) To keep using the already factorized L and U matrices as the preconditioner during nonlinear iteration, we apply the successive variable chord method [6] as an alternative to the Newton-Raphson method and improve the piecewise nonlinear definition of MOSFETs. The low-rank update technique is implemented to refresh the preconditioner efficiently. Further, incomplete L and U matrices can be derived to act as the preconditioner for the better performance.

With these techniques, the GMRES method is able to converge in a small number of iterations and the number of required LU factorizations is reduced dramatically.

This paper is organized as follows. Section 2 presents the improved piecewise nonlinear definition of MOSFETs. The new preconditioned GMRES method is proposed in Section 3. Section 4 summarizes experimental results on general nonlinear circuits and power/ground network examples. Finally, conclusions are drawn in Section 5.

2. PWNL definition of MOSFETs

In SILCA, a heuristic piecewise weakly nonlinear (PWNL) definition of MOSFETs was proposed. The purpose was to keep the circuit matrices constant when nonlinear iterations are performed within a PWNL region. Therefore, the number of LU factorizations can be reduced. In this section, we start from discussing the convergence property of the successive variable chord (SVC) method. Then systematic rules to generate the PWNL regions of MOSFETs are described. Finally, we briefly review the low-rank update technique.

Suppose that nonlinear iterations are performed within a PWNL region of a nonlinear function $f(x)$ to solve $f(x)=0$, as shown in Fig. 1, nonlinear iteration can be expressed by,

$$x_{i+1} = x_i - \frac{f(x_i)}{g} \tag{1}$$

where g is the chord for this PWNL region. Let the exact solution be $x^* = x_i + \varepsilon_i = x_{i+1} + \varepsilon_{i+1}$. Subtracting x^* from both sides of Eq. (1) gives,

$$\varepsilon_{i+1} = \varepsilon_i + \frac{f(x_i)}{g} \tag{2}$$

By the Taylor expansion of $f(x)$ at x_i, we obtain the following error estimation,

$$\varepsilon_{i+1} \approx \varepsilon_i \left(1 - \frac{f'(x_i)}{g}\right) - \varepsilon_i^{\,2} \frac{f''(x_i)}{2g} \tag{3}$$

Eq. (3) shows clearly that a quadratic convergence rate is achieved if g is equal to $f'(x_i)$, which is the Newton-Raphson method. Otherwise, the convergence rate is reduced to be linear, which is the case for the successive variable chord method. We observed that, on one hand, the smaller the $|1- f'(x_i)/g|$ is, the closer to the quadratic convergence rate Eq. (3) is. On the other hand, the larger the $|1- f'(x_i)/g|$ is, the larger the range of a PWNL region could be. Apparently, there exists a tradeoff between the convergence rate and the range of a PWNL region. We define the following condition with a parameter $\delta<1$,

$$\left|1 - \frac{f'(x_i)}{g}\right| < \delta \tag{4}$$

For a PWNL region as shown in Fig. 1, it can be derived from Eq. (4) that,

$$\frac{f'_{\max}}{1+\delta} < g < \frac{f'_{\min}}{1-\delta} \tag{5}$$

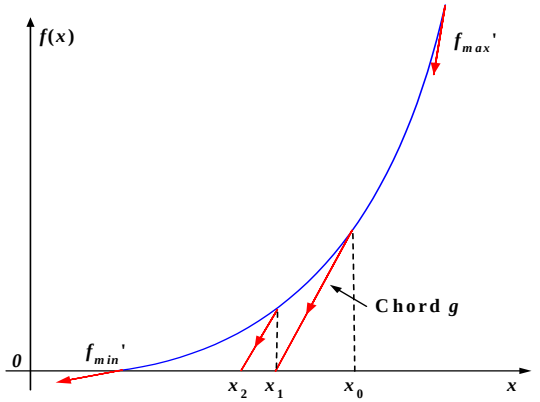

Figure 1. SVC method for $f(x)$ within a PWNL region.

It should be noted that the above analysis is done in the context that nonlinear iterations are performed within a PWNL region. In case that nonlinear iterations run across two or more PWNL regions, such as the example shown in Fig. 2 where the exact solution resides at the boundary of two PWNL regions, the following condition should be satisfied to achieve convergence,

$$x_2 > x_0 \tag{6}$$

Thus, g_1 and g_2 should satisfy the following inequality,

$$(g_1 - a) + (g_2 - a) > \frac{f(x_1) - f(x_0)}{x_1 - x_0} - a \qquad (7)$$

In our experiments, to satisfy both Eq. (5) and Eq. (7), the chord is chosen to be the maximum derivative in each PWNL region. With the knowledge of device model behaviors, such as monotonicity, the maximum derivative for each PWNL region can be computed.

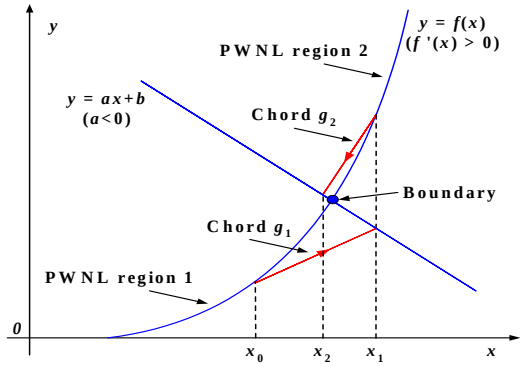

Figure 2. SVC method for $f(x)$ near the boundary of two PWNL regions.

It should be noted that PWNL regions of a nonlinear function is equivalent to piecewise constant (PWC) regions of the first-order derivatives of the nonlinear function. The following three rules can be used to generate PWNL regions for the MOSFET model.

1) The maximum voltages of V_{ds} and V_{gs} are predefined. In our experiments, we use V_{dd} as the maximum voltage for both of them. Given model parameters, the maximum g_{ds} and g_m can be calculated.
2) With a predefined δ<1, the PWC regions for g_{ds} and g_m are calculated as below,
$$g_n = g_{\max}$$
$$g_{i-1} = (1-\delta) g_i, \qquad i = n, n-1, ..., 2$$
3) A lower bound of g_{ds} and g_m is predefined, so that the rule (2) will stop whenever g_{ds} and g_m are less than the predefined lower bound. This is necessary to avoid a PWC region for g_{ds} and g_m being too narrow.

With the above rules, the PWC regions for g_{ds} and g_m of the MOSFET level 1 model in the two-dimensional V_{ds} and (V_{gs}-V_{th}) plane are shown in Figs. 3 and 4, respectively. There are five PWC regions for both g_{ds} and g_m.

It is clear from Figs. 3 and 4 that g_{ds} and g_m reach their maximum values in different PWNL regions. It should be noted that effects due to V_{bs} have been incorporated into V_{th}. For the MOSFET level 1 model, g_{mbs} has a simple relationship with g_m [13],

$$g_{mbs} = g_m * \frac{dV_{th}}{dV_{sb}} = g_m * \frac{\gamma}{2\sqrt{\Phi + V_{sb}}} \qquad (8)$$

For the simplification purpose, we always use the maximum dV_{th}/dV_{sb} in our experiments. Therefore, the PWC regions for g_{mbs} are the same as those for g_m. The rules of generating PWNL regions can be applied to complicated MOSFET models such as BSIM3 [13], as well.

As mentioned in [3], the low-rank update technique is an efficient method to update the factorized L and U matrices when only a few nonlinear devices change their PWNL regions. When a MOSFET switches its operating PWNL region, the contribution of this MOSFET to the circuit matrix changes as follows,

$$\begin{array}{c} \\ \mathrm{D} \\ \mathrm{S} \end{array} \begin{array}{c} \begin{array}{cccc} \mathrm{D} & \mathrm{G} & \mathrm{S} & \mathrm{B} \end{array} \\ \begin{bmatrix} \Delta g_{ds} & \Delta g_m & -\Delta g_{ds} - \Delta g_m - \Delta g_{mbs} & \Delta g_{mbs} \\ -\Delta g_{ds} & -\Delta g_m & \Delta g_{ds} + \Delta g_m + \Delta g_{mbs} & -\Delta g_{mbs} \end{bmatrix} \end{array}$$

$$= \begin{bmatrix} \sqrt{|a|} \\ -\sqrt{|a|} \end{bmatrix} \begin{bmatrix} \frac{\Delta g_{ds}}{\sqrt{|a|}} & \frac{\Delta g_m}{\sqrt{|a|}} & -\frac{a}{\sqrt{|a|}} & \frac{\Delta g_{mbs}}{\sqrt{|a|}} \end{bmatrix}, \quad a = \Delta g_{ds} + \Delta g_m + \Delta g_{mbs}$$

With the above representation, the low-rank update technique can be applied.

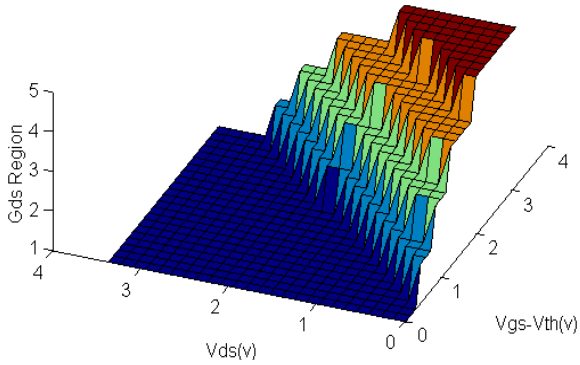

Figure 3. PWC regions of g_{ds} in two-dimensional V_{ds}-(V_{gs}-V_{th}) plane.

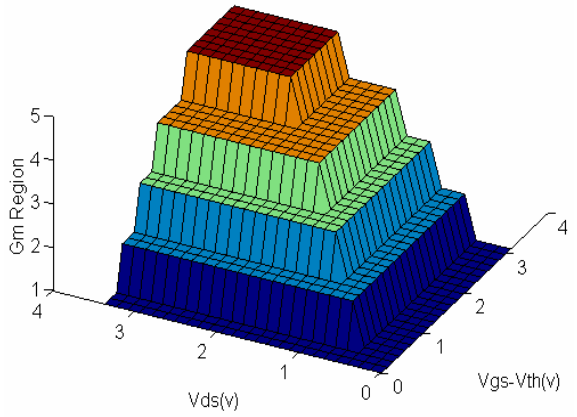

Figure 4. PWC regions of g_m in two-dimensional V_{ds}-(V_{gs}-V_{th}) plane.

3. Preconditioned GMRES method

The transient simulation flow of the proposed preconditioned GMRES method is shown in Algorithm I described below. It is clear from Algorithm I that LU factorizations are only performed when time step-sizes vary out of the predefined h_n/h range. In other cases, the L and U matrices are either kept constant or updated by the low-rank update technique when nonlinear devices switch their piecewise nonlinear regions. During the whole process, the L and U matrices are used for forward/

backward substitution (FBS) and act as the preconditioner for the GMRES method.

Three types of preconditioners are tested in our experiments: 1) The full L and U matrices. 2) Type I incomplete L and U (ILU) matrices approximated from the full L and U matrices – a matrix element $a(i,j)$ in the L or U matrix is removed if $|a(i,j)|<c\cdot|a(i,i)|$ and $|a(i,j)|<c\cdot|a(j,j)|$. 3) Type II incomplete L and U matrices approximated from the full L and U matrices – a matrix element $a(i,j)$ is removed if $|a(i,j)|<c\cdot\max(|a(*,j)|)$ in L or $|a(i,j)|<c\cdot\max(|a(i,*)|)$ in U. c is a small positive number, 0.001 is used in our experiments.

Algorithm I. Transient simulation flow.

```
DC operating point analysis
Choose an initial step size h0, the basis step size h = h0, t = 0
WHILE (t<Tfinal){
    OUTER LOOP: do{
        α = hn/h, iter_no = 0
        INNER LOOP: do{
            IF(0.625<α<2.5){
                IF(PWNL region is changed){
                    Apply low-rank update on L/U matrices
                }
                Apply preconditioned GMRES method
            }ELSE{
                IF(iter_no==0){
                    Apply LU factorization & FBS
                }ELSE{
                    IF(PWNL region is changed){
                        Apply low-rank update on L/U matrices
                    }
                    Apply FBS
                }
            }
            iter_no = iter_no + 1
        } while (not converged)
        Choose a new hn based on LTE requirement
    } while (LTE greater than predefined error limit)
    t = t + hn
}
```

In Algorithm I, the PWNL definition of MOSFETs described in Section 2 is used for the preconditioner. However, it is not necessary for the GMRES method to use the same PWNL definition to construct the circuit matrix equations. Instead, the GMRES method could still use standard MOSFET models so that accurate model information is included. This is especially useful for incorporating nonlinear capacitors, since nonlinear capacitors are generally simplified to linear ones when used for the preconditioner. Furthermore, by using standard MOSFET models for the GMRES method, the number of nonlinear iterations with the preconditioned GMRES method could be close to that with SPICE.

4. Experimental results

4.1 General nonlinear circuit examples

To verify the proposed GMRES method on general nonlinear circuits, several digital, analog and RF circuits have been tested and results are shown in Table I. The preconditioner for the GMRES method is the full L and U matrices. From Table 1, we see that the number of LU factorizations is reduced dramatically compared to that with SPICE. For the simplification purpose, we used the PWNL definition of MOSFETs for both the preconditioner and circuit matrix equations solved by the GMRES method. Therefore, the number of nonlinear iterations is generally increased to less than 2X of that with SPICE. Compared to SILCA, the proposed GMRES method generally requires less number of nonlinear iterations.

Table I. Simulation results on test circuits*.

Test Circuits	#Total points	#Accepted points	#Tran Iter	#Tran LU	#GMRES Iter
Inv	142	127	344	344	–
	141	127	380	63	253
20-stage inv chain	369	266	1193	1193	–
	357	259	2029	60	5275
Nand2	132	123	306	306	–
	120	112	421	54	324
One-shot trigger	501	421	1525	1525	–
	505	421	2650	198	5971
Comparator	145	127	444	444	–
	156	138	1071	60	1354
Opamp follower	19812	13816	74216	74216	–
	19723	13785	91808	11	219717
Ring oscillator	243	173	1022	1022	–
	260	192	2250	38	5186
VCO	1506	1045	7621	7621	–
	1600	1137	16096	399	47609

**Note: For each circuit, the 1st row is the SPICE3 result, and the 2nd row is the GMRES result.*

4.2 Power/ground network examples

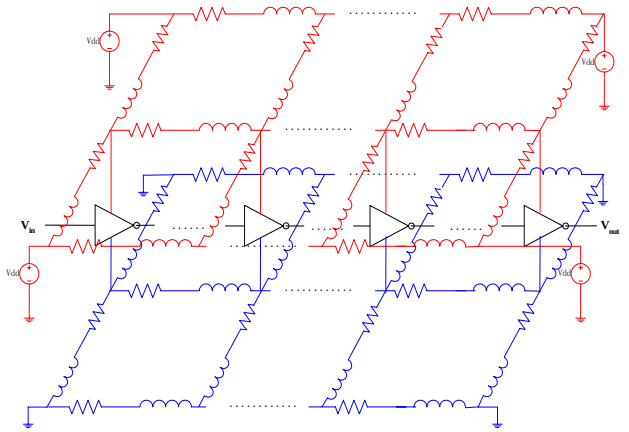

Figure 5. The power/ground network example.

To examine the efficiency of the proposed GMRES method, a power/ground network example as shown in Fig. 5 is simulated, which is similar to that used in [6]. The power and ground supply networks are modeled as two RCL mesh layers (parasitic coupling capacitors are not shown in Fig. 5). In our example, between these two layers is a 20-stage inverter chain, representing nonlinear circuits. Furthermore, RCL loads are added for each inverter to model interconnect lines between the adjacent stages. The

size of two RCL meshes can be changed to vary the number of elements.

Tables II, III and IV summarize the simulation results for power/ground network examples using the GMRES method with the full LU preconditioner, the type I ILU preconditioner and the type II ILU preconditioner, respectively. The error tolerance ε for the GMRES method is set to 1e-8. SPICE3 simulation results are also included in Table II. For clarity, the run time comparison is shown in Fig. 6. It is expected that more speedup could be achieved for larger power/ground networks.

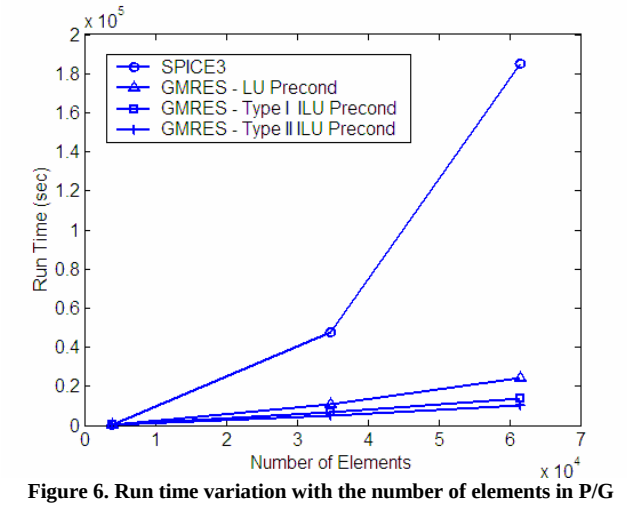

Figure 6. Run time variation with the number of elements in P/G network examples

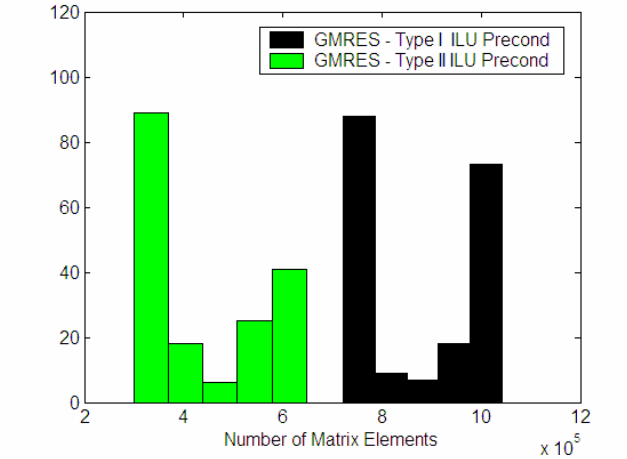

Figure 7. Histograms of the number of L and U matrix elements

It is seen that the GMRES method with the type II ILU preconditioner achieves the best speedup over SPICE3 for the largest power/ground network – 18.02*X*. The reason is that the number of matrix elements in the type II ILU preconditioner is much less than those in the LU preconditioner and the type I ILU preconditioner, especially for large matrices. For the power/ground network example with 4002 elements, the histograms of the number of L and U matrix elements during transient simulation are shown in Fig. 7. The number of matrix elements in full L and U matrices is 3116915 for this example. It is observed that the number of L and U matrix elements in the type II ILU preconditioner is reduced to about 1/10~1/5 of that in the full LU preconditioner.

With the error tolerance ε set to 1e-8, the average number of GMRES iterations in each GMRES solving process (#GMRES Iter / #GMRES) is about 3.20 to 3.35 for the GMRES method with the full LU preconditioner as shown in Table II. It increases to about 3.75 to 4.50 and 4.05 to 4.75 for the GMRES method with the type I ILU preconditioner (Table III) and the type II ILU preconditioner (Table IV), respectively. It is clear that the proposed preconditioner is efficient for the GMRES method during time-domain VLSI circuit simulation. In our experiments, when the error tolerance ε is further decreased to 1e-10 for higher accuracy, the average number of GMRES iterations in each GMRES solving process increases to about 6.60 to 8.35 for the GMRES method with the type II ILU preconditioner. As a result, the GMRES method requires more run time when the error tolerance is made smaller. Therefore, there exists a tradeoff between the accuracy and efficiency.

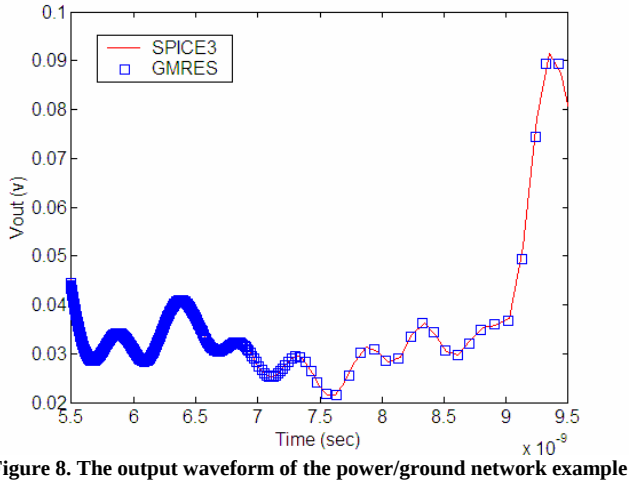

Figure 8. The output waveform of the power/ground network example

Figure 8 shows the output waveform of the inverter chain between the power and ground networks. It is seen that the low-level voltage of the output is larger than the ideal ground voltage due to the *IR* drop and *L*dI/dt* effects. It is clear in Fig. 8 that the accuracy with the proposed GMRES method is comparable to that with SPICE3.

It is worthy noting that the number of required LU factorizations (#Tran LU) is reduced dramatically with the preconditioned GMRES method compared to SPICE3 (#Tran LU = #Tran Iter). Furthermore, compared to the simulation results with SILCA, a similar speedup over SPICE3 has been achieved by the preconditioned GMRES method. However, the proposed preconditioned GMRES method is free of numerical stability problems as encountered in SILCA. Although the number of iterations (#Tran Iter) with the preconditioned GMRES method is increased to about 1.5*X* due to the PWNL definition of MOSFETs, it is still much less than that with SILCA.

5. Conclusion

In this paper, an efficiently preconditioned GMRES method is presented to speedup the transient simulation of parasitic-sensitive deep-submicron VLSI circuits. The cost of LU factorizations is minimized since they are only performed if time step-sizes vary violently. When time step-sizes change within a predefined range, the GMRES method is invoked with the preconditioner coming from the previously factorized L and U matrices. An improved PWNL definition of MOSFETs is also proposed to reduce the number of nonlinear iterations. With these techniques, the cost of required LU factorizations has been reduced dramatically. Orders of magnitude speedup has been achieved on power/ground network examples with the SPICE-like accuracy. The speedup could be further boosted by the parallel implementation of the GMRES method [11].

References

[1] E. Acar, F. Dartu, and L. T. Pileggi, "TETA: Transistor-Level Waveform Evaluation for Timing Analysis", *IEEE Trans. on Computer-Aided Design*, vol. 21, no. 5, pp. 605-616, May 2002.

[2] T. Chen and C. C.-P. Chen, "Efficient Large-Scale Power Grid Analysis based on Preconditioned Krylov-subspace Iterative Methods", *Proc. IEEE/ACM Design Automation Conference*, pp. 559-562, June 2001.

[3] T. Fujisawa, E. S. Kuh, and T. Ohtsuki, "A Sparse Matrix Method for Analysis of Piecewise-Linear Resistive Networks", *IEEE Trans. on Circuit Theory*, vol. CT-19, no. 6, pp. 571-584, November 1972.

[4] J. N. Kozhaya, S. R. Nassif, and F. N. Najm, "A Multigrid-like Technique for Power Grid Analysis", *IEEE Trans. on CAD*, vol. 21, no. 10, pp. 1148-1160, Oct. 2002.

[5] Y.-M. Lee and C. C.-P. Chen, "Power Grid Transient Simulation in Linear Time Based on Transmission-Line-Modeling Alternating-Direction-Implicit Method", *Proc. IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 75-80, Nov. 2001.

[6] Z. Li and C.-J. R. Shi, "SILCA: Fast-Yet-Accurate Time-Domain Simulation of VLSI Circuits with Strong Parasitic Coupling Effects", *Proc. IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 793-799, Nov. 2003.

[7] Z. Li and C.-J. R. Shi, "A Coupled Iterative/Direct Method for Efficient Time-Domain Simulation of Nonlinear Circuits with Power/Ground Networks", *Proc. IEEE Int. Symp. on Circuits and Systems*, pp. 165-168, May 2004.

[8] W. J. McCalla, *Fundamentals of Computer-Aided Circuit Simulation*, Kluwer Academic Publishers, 1988.

[9] L. W. Nagel, *SPICE: A Computer Program to Simulate Semiconductor Circuits*, University of California, Berkeley, Tech. Rep., UCB/ERL M520, May 1975.

[10] J. R. Phillips and L. M. Silveira, "Simulation Approaches for Strongly Coupled Interconnect Systems", *Proc. IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 430-437, November 2001.

[11] Y. Saad, "*Iterative Methods for Sparse Linear Systems*", 2nd Edition, SIAM, 2003.

[12] M. Zhao, R. V. Panda, S. S. Sapatnekar, and D. Blaauw, "Hierarchical Analysis of Power Distribution Networks", *IEEE Trans. on CAD*, vol. 21, no. 2, pp. 159-168, Feb 2002.

[13] Star-Hspice Manual, Release 1998.2, Avanti! Corporation, 1998.

Table II. Simulation results for the power/ground network example (ε=1e-8, LU preconditioner).

#Elems	SPICE3			Preconditioned GMRES							Speedup
	#Tran Iter	Tran LU (sec)	Tot Tran (sec)	#Tran Iter	#Tran LU	#GMRES	#GMRES Iter	Tran LU (sec)	GMRES (sec)	Tot Tran (sec)	
4002	4023	371.20	403.99	6086	49	5945	19872	4.42	114.12	132.48	3.05
34802	4006	4.549e4	4.760e4	7083	51	6922	22140	661.66	9572.16	10648.52	4.47
61602	4377	1.797e5	1.848e5	7207	63	7000	22476	2848.69	20549.69	24275.38	7.61

Table III. Simulation results for the power/ground network example (ε=1e-8, type I ILU preconditioner).

#Elems	Preconditioned GMRES							Speedup
	#Tran Iter	#Tran LU	#GMRES	#GMRES Iter	Tran LU (sec)	GMRES (sec)	Tot Tran (sec)	
4002	6682	46	6547	24710	4.01	90.37	108.98	3.71
34802	7017	55	6836	29143	696.01	5538.22	6637.26	7.17
61602	6624	49	6462	29012	2123.18	10602.02	13458.90	13.73

Table IV. Simulation results for the power/ground network example (ε=1e-8, type II ILU preconditioner).

#Elems	Preconditioned GMRES							Speedup
	#Tran Iter	#Tran LU	#GMRES	#GMRES Iter	Tran LU (sec)	GMRES (sec)	Tot Tran (sec)	
4002	6771	53	6618	26886	4.72	79.11	99.08	4.08
34802	6682	54	6509	29746	680.48	3674.44	4740.55	10.04
61602	6758	52	6586	31354	2221.41	7249.35	10252.93	18.02