Proc. NUPADIV, Seattle, WA, 5/92

Waveform Frequency-Dependent Overrelaxation for Transient Two-Dimensional Simulation of MOS Devices

Mark W. Reichelt Jacob White Jonathan Allen

Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, MA 02139, USA

Abstract

In this paper a new waveform frequency-dependent overrelaxation algorithm is presented and applied to solving the differential-algebraic system generated by spatial discretization of the time-dependent semiconductor device equations. In the experiments included, the frequency-dependent overrelaxation method converged robustly and was up to 7 times faster than ordinary waveform relaxation.

1 Introduction

The enormous computational expense and the growing importance of mixed circuit/device simulation, as well as the increasing availability of parallel computers, suggest that specialized, easily-parallelized algorithms be developed for the transient simulation of MOS devices [1]. Recently, the easily-parallelized waveform relaxation (WR) algorithm was shown to be a computationally efficient approach to device transient simulation [2], even though the device WR algorithm typically requires hundreds of iterations to achieve an accurate solution. In this paper we investigate a frequency-dependent overrelaxation scheme for the WR algorithm and present experimental results which demonstrate that this scheme converges robustly and reduces the number of iterations by up to a factor of 7.

2 Ordinary Waveform Overrelaxation

Device transient simulation is usually performed by numerically solving the coupled Poisson and time-dependent electron and hole current-continuity equations with a low-order implicit time-integration scheme, combined with a Newton or relaxation method to solve the generated sequence of nonlinear algebraic equations [3, 4]. Another approach is to apply WR and standard overrelaxation acceleration (WSOR) to the equation system, as given in Algorithm 1 [2, 5, 6]. Unfortunately, even with a carefully chosen overrelaxation parameter, the ordinary WSOR algorithm can produce oscillatory results, as illustrated in Figure 1.

Algorithm 1 (Ordinary WSOR for Device Simulation)

guess $\boldsymbol{u}^0, \boldsymbol{n}^0, \boldsymbol{p}^0$ `waveforms at all nodes`
`for` $k = 0, 1, \ldots$ `until converged`
  `for each node` i
    `solve for` $u_i^{k+1}, n_i^{k+1}, p_i^{k+1}$ `waveforms (as in Gauss-Seidel WR):`

$$\begin{aligned} f_{1_i}(u_i^{k+1}, n_i^{k+1}, p_i^{k+1}, u_j^k) &= 0 \\ \frac{d}{dt} n_i^{k+1} + f_{2_i}(u_i^{k+1}, n_i^{k+1}, u_j^k, n_j^k) &= 0 \\ \frac{d}{dt} p_i^{k+1} + f_{3_i}(u_i^{k+1}, p_i^{k+1}, u_j^k, p_j^k) &= 0 \end{aligned}$$

    `overrelax` $u_i^{k+1}, n_i^{k+1}, p_i^{k+1}$ `waveforms` $x_i^{k+1} \leftarrow x_i^{k+1} + \alpha \cdot \left[x_i^{k+1} - x_i^k \right]$

3 Waveform Frequency-Dependent Overrelaxation

To derive a more reliable acceleration, the effect of each iteration of Gauss-Seidel WR can be represented abstractly as

$$\boldsymbol{x}^{k+1} = F(\boldsymbol{x}^k) \quad \text{where} \quad F : C^1 \to C^1, \tag{1}$$

C^1 is the space of continuously differentiable functions, and $\boldsymbol{x}^k(t) = [\boldsymbol{u}^k(t), \boldsymbol{n}^k(t), \boldsymbol{p}^k(t)]^T$. Then if $\Delta^k(t) = \boldsymbol{x}^{k+1}(t) - \boldsymbol{x}^k(t)$ is small, a linearization of equation (1) followed by Fourier transformation yields

$$\Delta^{k+1}(\omega) = G(\omega)\, \Delta^k(\omega) \tag{2}$$

where $G(\omega)$ is the Fourier transform of the linearized F. The WR algorithm can then be accelerated in the frequency domain by overrelaxing:

$$x_i^{k+1}(\omega) \leftarrow x_i^{k+1}(\omega) + \alpha(\omega) \cdot \left[x_i^{k+1}(\omega) - x_i^k(\omega) \right] \tag{3}$$

Inverse transformation yields the following time-domain overrelaxation expression:

$$x_i^{k+1}(t) \leftarrow x_i^{k+1}(t) + \int_{-\infty}^{\infty} \alpha(t-\tau) \cdot \left[x_i^{k+1}(\tau) - x_i^k(\tau) \right] \, d\tau \tag{4}$$

With the waveform frequency-dependent successive overrelaxation algorithm (WFD-SOR), instead of multiplying the delta in the time domain by a constant parameter α as in ordinary WSOR, the delta in the frequency domain is multiplied by a frequency-dependent overrelaxation parameter $\alpha(\omega)$. The rationale for this approach is that different frequency components of $\boldsymbol{x}$ converge at different rates. In practice, the frequency-dependent overrelaxation parameter $\alpha(\omega)$ is computed as follows:

Step 1 Perform enough initial Gauss-Seidel (GS) WR iterations so that the largest eigenvalue $\gamma(\omega)$ of the GS WR operator dominates convergence at each frequency.

Step 2 Estimate the largest magnitude eigenvalue $\gamma(\omega)$ of the GS WR operator with the Rayleigh quotient:

$$\gamma(\omega) = \frac{[\Delta^k(\omega)]^*\Delta^{k-1}(\omega)}{[\Delta^k(\omega)]^*\Delta^k(\omega)}$$

Step 3 Compute the overrelaxation parameter: $\alpha(\omega) = \dfrac{2}{1+\sqrt{1-\gamma(\omega)}} - 1$

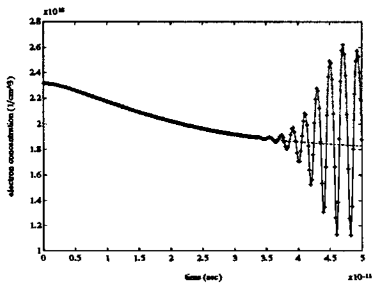

Figure 1: Electron concentration vs. time at a channel node of the **karD** example, showing frequency amplification resulting from ordinary WSOR (solid).

4 Experimental Results

The WFDSOR method was implemented in the WR-based device transient simulation program WORDS [2]. WORDS uses red/black block Gauss-Seidel WR, where the blocks correspond to vertical mesh lines. The equations governing nodes in the same block are solved simultaneously using the second-order backward-difference formula. The implicit algebraic systems generated by the backward difference formula are solved with Newton's method, and the linear equation systems generated by Newton's method are solved with sparse Gaussian elimination.

The four MOS devices of Table 1 were used to construct eight simulation examples, each device being subjected to either a drain voltage ramp with the gate held high (the **D** examples), or a gate voltage ramp with the drain held high (the **G** examples). All examples ranged from low to high drain current, and in the **G** examples, the gate displacement current was substantial because the applied voltage ramps changed at a rate of $.2 \sim 2$ volts per picosecond. The drain-driven **karD** test setup is illustrated in Figure 2.

Figure 3 shows the convergence of the eight examples as a function of iteration for WR, ordinary WSOR, and WFDSOR. The overrelaxation algorithms began with 64 initial WR iterations, after which, the WFDSOR parameter $\alpha(t)$ was calculated as described above, and the WSOR constant parameter α was calculated by estimating the convergence rate of WR in the time domain. For these plots, convergence was measured *a posteriori* as the maximum of the terminal current error, including both resistive and displacement currents. The WR, WSOR and WFDSOR algorithms were run for 500 iterations, regardless

device	description	L	L_{eff}	t_{ox}	mesh	unknowns
jar	empirical profile	0.27	0.17	5.2	24×31	1852
kar	abrupt junction	3.0	2.2	50	19×31	1379
ldd	lightly-doped drain	0.64	0.4	19	18×24	656
soi	silicon-on-insulator	0.7	0.5	7	18×24	656

Table 1: Description of MOS devices: gate length L (μm), effective channel length L_{eff} (μm), oxide thickness t_{ox} (nm), mesh size (rows×cols), and total number of u, n and p unknowns. Silicon thickness of the **soi** device is $t_{si} = 0.1$ μm.

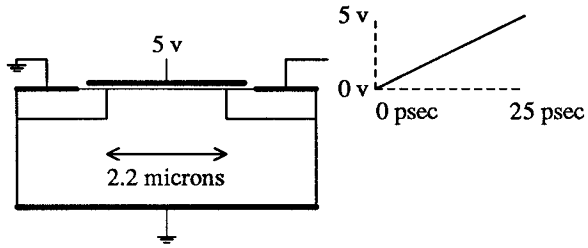

Figure 2: The drain-driven **karD** example.

of whether convergence to an accuracy tolerance was achieved earlier. To further simplify comparisons, 64 equally-spaced timesteps were used in all experiments.

The convergence of the WFDSOR algorithm for all examples demonstrates its robustness. The **jarD** and **jarG** examples do not converge for ordinary WSOR because the overrelaxation in the time domain amplifies some frequencies. The **karD** and **karG** examples also displayed frequency amplification during ordinary WSOR. The **lddD** and **lddG** examples converge better for ordinary WSOR than for WFDSOR, perhaps because the overrelaxation parameter $\alpha(\omega)$ was computed immediately after the initial 64 WR iterations, and was not updated to fit subsequent convergence.

example	unknowns	WR	WSOR	WFDSOR
jarD	1852	2308	—	315
jarG	1852	1510	—	334
karD	1379	332	95	105
karG	1379	288	89	105
lddD	656	321	100	188
lddG	656	224	83	150
soiD	656	151	79	92
soiG	656	154	79	94

Table 2: Number of iterations required by WR, WSOR and WFDSOR to reduce the max-norm error of the terminal currents (including displacement current) below 0.1%.

Table 2 compares the number of iterations required by WR, WSOR and WFDSOR to reduce the max-norm error of the terminal currents (including displacement current) below

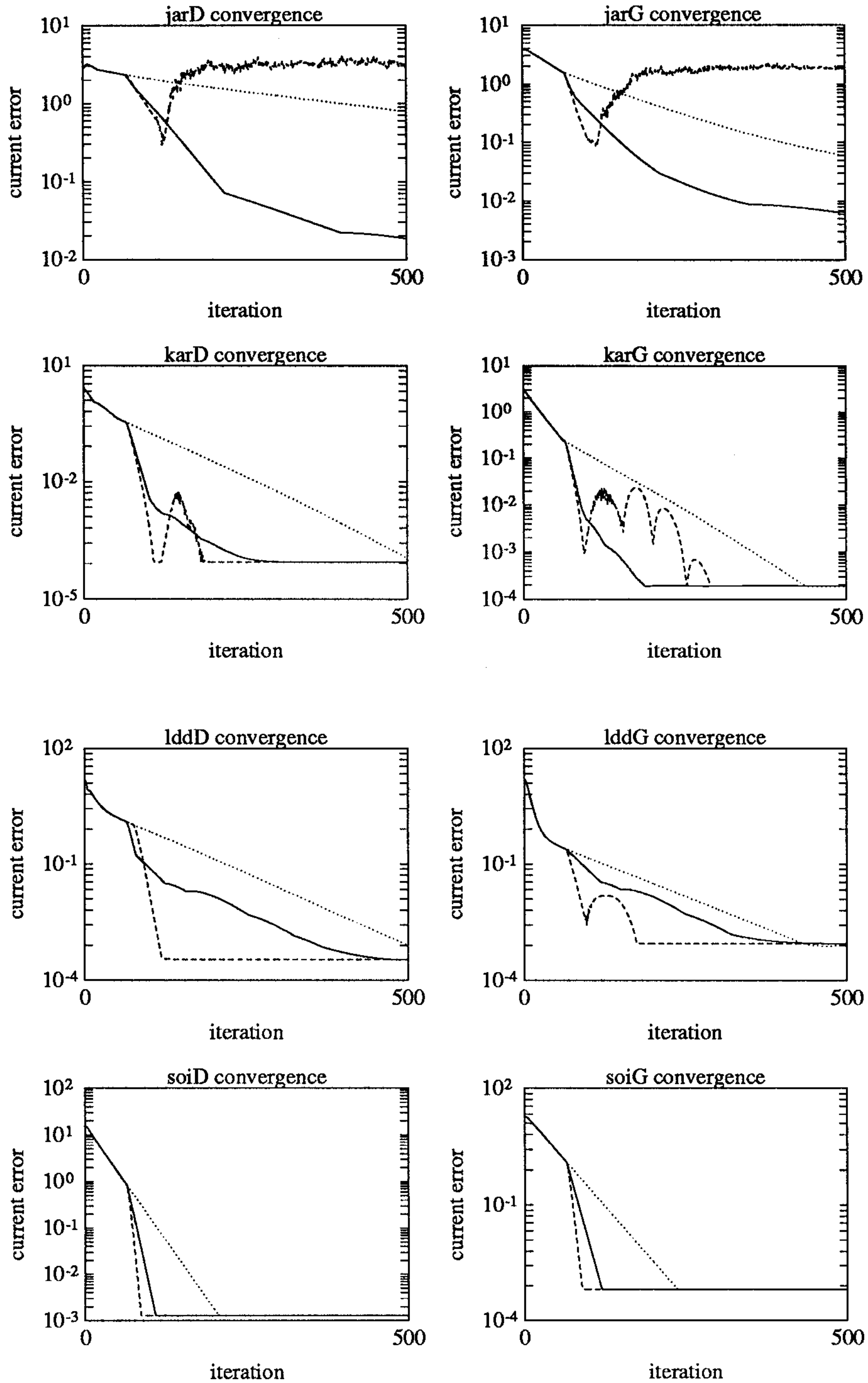

Figure 3: Maximum terminal current error of the eight examples as a function of iteration for WR (dotted), ordinary WSOR (dashed), and WFDSOR (solid).

0.1%. The table indicates that the WFDSOR acceleration reduced the number of WR iterations by up to a factor of 7.

5 Conclusion and Future Work

In this paper, a new waveform frequency-dependent overrelaxation algorithm was presented and applied to solving the differential-algebraic system generated by spatial discretization of the time-dependent semiconductor device equations. In the experiments included, the frequency-dependent overrelaxation algorithm converged robustly up to 7 times faster than ordinary WR.

Future work is focused first on improving the computation of the overrelaxation waveform $\alpha(t)$. It may be possible to compute $\alpha(t)$ more accurately in the time domain, avoiding any transformation. This will also make the transition to a non-fixed timestep algorithm easier. Finally, the $\alpha(t)$ will be adjusted adaptively, to assure good convergence.

The authors would like to thank Dennis Freeman, Ibrahim Elfadel and others of RLE for valuable discussions. This work was supported by a grant from IBM, the Defense Advanced Research Projects Agency contract N00014-91-J-1698, and the National Science Foundation contract MIP-8858764 A02.

References

[1] W. L. Engl, R. Laur, and H. K. Dirks, "MEDUSA – A simulator for modular circuits," *IEEE Trans. Computer-Aided Design*, vol. CAD-1, pp. 85–93, April 1982.

[2] M. Reichelt, J. White, and J. Allen, "Waveform relaxation for transient two-dimensional simulation of MOS devices," in *International Conference on Computer-Aided Design*, (Santa Clara, CA), pp. 412–415, November 1989.

[3] R. E. Bank, W. C. Coughran, Jr., W. Fichtner, E. Grosse, D. Rose, and R. Smith, "Transient simulation of silicon devices and circuits," *IEEE Trans. Computer-Aided Design*, vol. 4, pp. 436–451, October 1985.

[4] K. Mayaram and D. Pederson, "CODECS: A mixed-level device and circuit simulator," in *International Conference on Computer-Aided Design*, (Santa Clara, California), pp. 112–115, November 1988.

[5] E. Lelarasmee, A. E. Ruehli, and A. L. Sangiovanni-Vincentelli, "The waveform relaxation method for time domain analysis of large scale integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. 1, pp. 131–145, July 1982.

[6] U. Miekkala and O. Nevanlinna, "Convergence of dynamic iteration methods for initial value problems," *SIAM J. Sci. Statist. Comput.*, vol. 8, pp. 459–482, July 1987.