應用信號流強度於超大型積體電路之電路模擬

Circuit Simulation for Very Large-scale Integrated Circuits using Strength of Signal Flow

計畫編號: NSC90-2215-E-034-001

執行期限: 90 年 8 月 1 日至 91 年 7 月 31 日

主持人: 陳俊榮 中國文化大學資訊科學系

Email: chrischn@ms32.url.com.tw

一、摘要

本計劃的目的在解決超大型積體電路（VLSI)的線路模擬問題(Problem of Circuit Simulation for VLSI)，目前已經有許多演算法可以有效解決這個問題，我們在此要提出的做法是基於一種新的基礎，而求得新一代的模擬技術，這個新的基礎就是計算電路中各個電路變數間的影響敏感度，稱之為信號流強度（Strength of Signal Flow, SSF）的資訊，本計畫利用這個資訊設計出智慧型的新模擬技術，改善大型線路模擬的效能。本計劃有兩個基本工作，其一是有效計算信號流強度，其二則是設計能夠善加利用信號流強度的新演算法。本計劃具體的成果是製作可以實際模擬大型線路的智慧型線路模擬程式，可以模擬數個 VLSI 的範例電路，並將之和傳統的模擬程式做效能上的比較。

大型積體電路的線路模擬問題有許多有效的解決方法，基於鬆弛法（Relaxation-based）系列的演算法是其中的佼佼者，我們發現此類方法之發展已經頗為成熟，要再做更進一步的改善，必須引入新的元素，我們要加入的新元素便是信號流強度。加入信號流量強度的好處是可以發現各個電路變數間的相依程度；我們知道基於鬆弛演算法的基本做法都是利用此相依程度資訊，如利用它以分割子線路（Circuit Partitioning），利用它以決定子線路模擬之順序（Scheduling)。能夠精確掌握動態的信號流強度可以為基於鬆弛演算法的效能帶來極大的助益。

利用信號流強度的方式有許多種，最直接的便是動態地依據信號流強度改變線路的分割，保證模擬時都是用最佳的分割；再者是做完整個模擬流程後計算出這個模擬的最佳靜態子線路分割，再一次模擬此線路時可以有較佳的模擬效果（這個做法可以用在線路最佳化時對相似電路的多次模擬場合)；最後一種應用是將信號流強度融入基於鬆弛法演算法中以控制子線路的計算順序，來提昇計算效能。

預計本專案完成後可以得到一個大型電路模擬程式，其具有信號流強度計算功能、有可以善加利用信號流強度的基於鬆弛演算法、和計算靜態子線路分割的功能，這個程式將可以處理許多的測試電路，預計可以有較一般 VLSI 線路模擬程式較佳的效能。

Abstract

In circuit simulation, Relaxation-based algorithms have been proven to be faster and more flexible than the standard direct approach used in SPICE. Signal flow of the simulated circuit is very important in using Relaxation-based algorithm. However, there is no specific research undertaken for it. This paper defines the strength of signal flow (SSF) in circuit simulation, and discusses how to calculate it. Since SSF has crucial effects in Relaxation-based algorithms, this paper also devises techniques to utilize SSF in Relaxation-base algorithms for improving efficiency. Experimental examples on digital as well as analog circuits are given to prove the value of utilizing SSF in circuit simulation.

二、計畫緣由與目的

本計劃的目的是希望就超大型積體電路之線路模擬問題改進其效能，推出使用額外資訊的新一代的超大型積體電路線路模擬演算法，以更有效率地求解這種問題。

大型電路之線路模擬問題在過去數年間已經獲得相當程度的解決，有許多方法可以有效解決這個問題[1-6]。基於鬆弛法（Relaxation-based）[2]系列的演算法如ITA(Iterated Timing Analysis)[3]，WR(Waveform Relaxation)[2]和STWR（Selective-tracing Waveform Relaxation）[7]演算法是其中的佼佼者，此類演算法使用最精確的模擬模型(simulation model)，求解出如同標準線路模擬器 SPICE 水準的精準解答；其他的模擬法有使用簡化模擬模型的方法，如使用片段線性連續模型（Piecewise Linear Model）[4, 6]，和簡化 RC 電路模型[5]等等，此類方法在簡化和效率間做不同程度的妥協，所得結果較標準線路模擬器的結果要粗糙，但執行速度較快；各種方法在不同的需求場合各有其市場。

使用基於鬆弛演算法可以求解得和標準線路模擬程式 SPICE 相同程度精確的解答，而且其效能頗佳，在求得相當精確度的解答前提下可以有較 SPICE 一到兩個 factor 的時間節省，表現相當不錯，所以本計劃便鎖定此類演算法，問題是此類方法之發展已經頗為成熟，如果要再做更進一步的改善，必須引入新的元素，加入新的參考資訊，也就是將此種方法更進一步發展使成為新一代的演算法，最後能夠使用智慧的方式做線路模擬計算，而這也正是本計劃的目的。

探討如何進一部改進基於鬆弛演算法的計算效能，我們的做法是加入新的元素：信號流強度。本計畫的目的可以描述如下：

（1） 探討信號流強度的計算

（2） 求取子線路的信號流強度

（3） 在線路模擬演算法中利用信號流強度

（4） 獲得實際執行效能改進的數據

在下面的小節中，依序說明這些目的的完成結果。

三、結果與討論

本計畫的執行可說是圓滿的，而且已有數篇論文發表[8][9][10]。我們依序看看所希望達成的目的之執行成效，其中引用所發表論文中的內容。

（1） 探討信號流強度的計算

（2） 求取子線路的信號流強度

已經成功的找出信號流強度的計算方法：

SSF is the degree of influence between two circuit variables, in which one circuit variable is the affecting one and another is the affected one. It's straightforward to define SSF as the differentiation on the affected circuit variable with respect to the affecting circuit variable. Such definition is similar to that for sensitivity [3]. Therefore, the calculation for SSF is similar to that for sensitivities [3]. Assume there are two node/branch *a* and *b*, and we discuss the degree of influence from *a* to *b*:

$$SSF(a,b)=\frac{\partial V_b}{\partial V_a} \tag{1}$$

SSF(a, b) is the strength of signal flow from node/branch *a* to node/branch *b*, where V_a and V_b are circuit variables associated with *a* and *b* respectively. We go forwards to see how it is calculated. The simulated circuit is described as follows:

$$F(Y(t),\dot{Y}(t),t)=0 \tag{2}$$

where *Y* is the vector of circuit variables, *t* is the time, *F* is a continuous function and "." means differentiation with respect to time. Since the Relaxation-based algorithm is used, (2) is partitioned into subcircuits, the *i*th one of which, say *a*, is:

$$F_i(Y_i(t),\dot{Y}_i(t),D_i(t),\dot{D}_i(t),t)=0 \tag{3}$$

The equation for *a* is rewritten as the abbreviate form:

$$f(y(t), \dot{y}(t), w(t), \dot{w}(t), t) = 0 \quad (4)$$

where y (Y_i, a subvector of Y) is vector of circuit variables in a, w (D_i, the decoupling vector) is the vector of circuit variables not in a, and f is a continuous function. Since we want to know all SSF of nodes in a with respect to an "input" circuit variable m in w, equation (4) is rewritten as:

$$f(y(m,t), \dot{y}(m,t), \hat{w}(t), \dot{\hat{w}}(t), m, t) = 0 \quad (5)$$

where $\hat{w}$ is the resulted vector of removing m from vector w. Differentiating (5) with respect to m, we have:

$$f_y \frac{\partial y}{\partial m} + f_{\dot{y}} \frac{\partial \dot{y}}{\partial m} + f_m = 0 \quad (6)$$

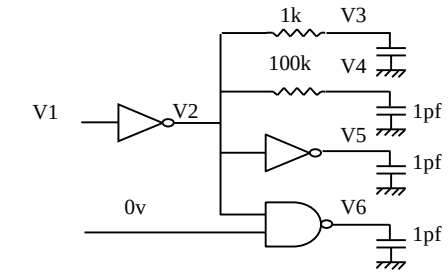
We use Trapezoidal rule to discretize this equation:

$$j_y s_{n+1} = (\frac{2}{h} f_{\dot{y}} s_n + f_{\dot{y}} \dot{s}_n) - f_m = O_n - f_m \quad (7)$$

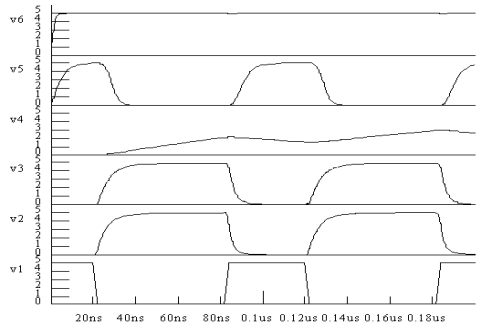
Where $s_{n+1} = \frac{\partial y(t_{n+1})}{\partial m}$ is the SSF vector of all nodes in a with respect to m (t_{n+1} is the current time point), $j_y = (\frac{2}{h} f_{\dot{y}} + f_y)$ is the Jacobian of (5), O_n is the vector collecting "old" values at previous time point t_n, and f_m is jth column of $j_w = \frac{\partial f}{\partial w} = (\frac{2}{h} f_{\dot{w}} + f_w)$ (assume that m is the jth element of w).

Equation (7) is the computing equation used to calculate transient SSF. We note that if m is the circuit variable of a node in one preceding subcircuit of a, say b, $|s|_\infty = |SSF(m, y)|_\infty$ (note that other kinds of norm can be applied, too) can be used to indicate the degree of influence from b to a (via circuit variable m). So, the strength of signal flow from b to a can be known. We note that (7) is used to calculate the influence from fan-in subcircuits rather than the influence to fan-out subcircuits.

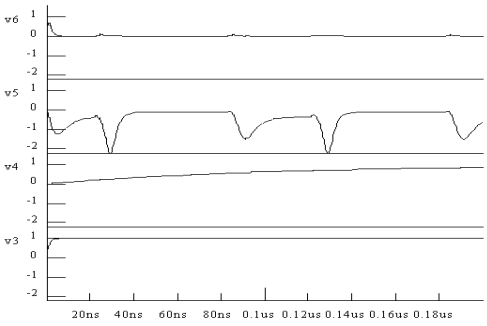
Fig. 1 demonstrates an example for SSF calculation. Fig. 1(a) describes circuit schematic, Fig. 1(b) contains the timing waveforms, and Fig. 1(c) contains waveforms of all SSF with respect to V2. We can find SSF of V3 and V4 with respect to V2 both become constant 1, for a resistor has good signal flow "conductance." The SSF of V5 with respect to V2 depends on operation states of transistors, in which SSF exists only when the NOT gate is toggling. The NAND gate has been turned off, so SSF of V6 is zero.



(a)



(b)



(c)

Fig. 1 (a) Schematic. (b) Timing waveforms. (c) SSF waveforms with respect to V2.

（3） 在線路模擬演算法中利用信號流強度

原先的計畫是在搭配信號流強度於某線路模擬演算法，所使用的演算法是 Iterated Timing Analysis (ITA)，我們改進了 ITA 的 Scheduling Scheme，使得任何子線路的排入（排入佇列中以便計算）都經過信號流強度的核可，如此，可以大幅提升演算法的效能。另外在應用信號流強度於動態子線路分割部分，我們也在 ITA 演算法中加入了 Dynamic Merging Scheme，使得線路的分割可以動態的改變。以下列出所得的演算法：

Algorithm 1 (ITA Algorithm Adopting SSF):

```
/* Simulation duration is T_begin ~T_end */
// E() is an priority queue, whose elements are queues
Put subcircuits connected to primary input into E(T_begin);
while(E is not empty) {
    t_n+1 = the smallest event-time in E;
    for(k = 1; E(t_n+1) is not empty; k++) {
        // k is the relaxation index
        Clear TMP;     // TMP is a queue
        for(each subcircuit a in E(t_n+1)) {
            Solve a at t_n+1 for transient responses;
LB3:        Add member subcircuits of a into L;
            // if a is not a merged subcircuit, a itself is the
            // only member subcircuit
LB4:        while(L is not empty) {
                Delete a from L;
                if(a has been converged) { // converged
                    Estimate next solving time t_next ;
                    add a into E(t_next);
                    // if a has been converged, calculate
                    // its SSF with respect to all input variables
MK:                 for(each input variables m of a) {
                        if(m is not in any subcircuit) continue;
                        Use (7) to calculate SSF of y with respect to
                            m, s_n+1;
                        Store s_n+1;
                    }
                }
                else {   // not converged
                    Add a into TMP;
LB1:                Add fan_out_ssf(a) into E(t_n+1);
LB2:                Find z∈ fan_out_ssf(a) , which has big SSF to a;
                    if(z exists) merge a and z together, delete a from
                        TMP;
                }
            }   // end of while(L is not empty)
        }
        E(t_n+1) = TMP;
    }
}
```

（4） 獲得實際執行效能改進的數據

以上的演算法確實可以大幅的縮減計算的時間，我們將之實作成線路模擬程式，然後用以分析許多電路上，紀錄分析的效能數據，驗證了它的好處，Table I 和 Table II 是所測試的電路規格和執行時間的列表。可以很清楚的觀察到模擬效能的改進。

TABLE I

SPECIFICATIONS OF SIMULATED CIRCUITS

Ckt	Name	Node #	MOSFET #	Subcircuit #	Simulation Duration
1	8-stage InvChain	80	160	80	200ns
2	4-bit ALU	200	400	112	100ns
3	4-bit SynCounter	88	176	44	400ns
4	4-bit RippleCounter	76	152	36	400ns
5	16-bit Shift Register	64	96	32	60ns
6	64X4 SRAM	1926	4492	898	50ns

TABLE II

USED CPU TIME AND COUNTS OF SUBCIRCUIT CALCULATIONS

Circuit	Used CPU Time		# Subckt Calculations		Over-head
	ITA	ITA+SSF	ITA	ITA+SSF	
8-stage InvChain	88.4	15.6	1,682,970	164,859	6.9
4-bit ALU	18.2	10.9	379,058	138,676	4.2
4-bit SynCounter	26.4	11.1	504,944	126,027	4.5
4-bit Ripple Counter	18.5	10.5	350,509	92,847	5.6
16-bit Shift Register	130	72.8	2,742,974	749,334	37.3
64X4 SRAM	119.6	82.5	837,672	467,462	15.8

CPU time is in Pentium III-550 seconds.

四、計畫成果自評

主持人和同學們都很認真的執行此計畫，本計畫原定的目標可說都完成了，我們也發表了數篇論文，還遠渡崇洋去日本法國參加研討會發表之。在執行此計畫的同時，經過了多次的討論，之後，又發現了信號流強度的可能應用方向，比如敏感度的計算似乎可以應用信號流強度簡化之，我們覺得這算是一個不錯的收穫。此外，我們覺得信號流強度在線路模擬方面的應用還可以加強，比如將之應用在 Waveform Relaxation (WR)演算法中，這些新的目標將在新年度的國科會計劃中加以研究。

五、參考文獻

[1] A. R. Newton and A. L. Sangiovanni-Vincentelli, "Relaxation-based electrical simulation, " IEEE Trans, Computer-aided Design, Vol. CAD-3, pp. 308-311, Oct. 1984.

[2] 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. CAD-1, pp. 131-145, Aug. 1982.

[3] R. A. Saleh and A. R. Newton, "The exploitation of latency and multirate behavior using nonlinear relaxation for circuit simulation," IEEE Trans., Computer-aided Design, vol. 8, pp. 1286-1298, December 1989.

[4] T. V. Nguyen, A. Devgan, O. J. Nastov, and D. W. Winston, "Transient sensitivity computation in controlled explicit piecewise linear simulation,"

IEEE Trans., Computer-aided Design, Vol. 19, NO. 1, pp. 98-110, Jan. 2000.

[5] R. Kay, and L. T. Pileggi, "EWA: Efficient wiring-sizing algorithm for signal nets and clock nets," IEEE Trans., Computer-aided Design, VOL. 17, NO. 1, pp. 40-49, January, 1998.

[6] Andrew R. Conn, Paula K. Coulman, Rund A. Haring, Gregory L. Morrill, Chandu Visweswariah, and Chai Wah Wu, "JiffyTune: Circuit Optimization using time-domain sensitivities", IEEE Tran. on Computer-aided Design, Vol. 17, No. 12, pp. 1292-1309, Dec. 1998.

[7] C. J. Chen, and W.S. Feng, “Transient sensitivity computations of MOSFET circuits using Iterated Timing Analysis and Selective-tracing Waveform Relaxation,” Proceeding of 31st Design Automation Conference, pp. 581-585, San Diego CA, June 1994.

[8] Chun-Jung Chen “Calculation for Strength of Signal Flow and Its Application in Circuit Simulation Algorithm,” 全國計算機會議, 台北, Dec. 2001.

[9] Chun-Jung Chen, and Wen-Pin Tai, “Signal flow analysis and its utilization in Relaxation-based circuit simulation,” The International Conference on Fundamentals of Electronics, Communications and Computer Sciences, Tokyo Japan, March 2002.

[10] Chun-Jung Chen, Wen-Pin Tai, and Jenn-Dong Sun, “Strength of signal flow in circuit simulation and its application,” Symposium on Design, Test, Integration, and Packaging of MEMS/MOEMS, Canne France, May 2002.