

数式処理を用いた LSI 設計に関する研究

A study of Symbolic Analysis for LSI design

中林太美世 加古富志雄

奈良女子大学

Tamiyo NAKABAYASHI Fujio KAKO

Nara Women's University

概要

本稿では、数式処理の応用として、数式処理ソフト Mathematica[7] を用いた論理回路設計について議論する。CMOS 非線形遅延モデルによる論理回路の遅延計算を数式処理を用いて行う方法を提案する。まず、基本セルの遅延特性を近似多項式の数式表現によりモデル化する。さらに、本手法を用いて 1 ビット全加算回路を設計し、配線長の最適化を行った。これらにより、本手法の有用性を確認した。

1 はじめに

数式処理の LSI 設計分野への適用について提案する。具体的なアプリケーションとして、LSI の信号遅延計算ツールの開発を目指す。

LSI 設計において回路が正常動作することを保証するためには、回路遅延の見積もりと最適化が必須である。LSI の高速化・高集積化を図るために製造工程の微細化と多層配線構造化が進み、信号配線の寄生容量／抵抗が信号遅延に及ぼす影響が大きくなっている [2][3][4]。一般的に、信号遅延を計算するための方法として、回路のカテゴリに応じて回路方程式を数値的に解くシミュレーション手法や解析式を用いる方法、あるいは予め Characterize しておいた特性テーブルを用いる方法がある。本研究の趣旨は、回路遅延の見積もり数式処理を適用することにより、信号遅延のモデル式による計算を通じて信号遅延の理論と過渡現象の概念を結び付けることである。

本報告では、研究の初期段階として、数式処理ソフト Mathematica を用いた CMOS 非線形遅延モデルによる論理回路の遅延計算手法を提案する。まず基本セルの遅延特性についてモデル化を行い、さらに実回路への適用例として 1 ビット全加算回路の設計を行った。その結果、回路動作や信号遅延のモデル式を使って数式ベースで最適な設計値を求めることができ、提案手法の有用性を確認した。

2 数式処理を用いた論理回路の遅延計算

CMOS 非線形遅延モデルを用いた論理回路の遅延計算を数式処理を用いて実現する方法を検討する。

2.1 CMOS 非線形遅延モデル

一般的に、論理回路の遅延計算は CMOS 非線形遅延モデルを用いて以下のように行われる[5]。CMOS 非線形遅延モデルは、予め Characterize しておいた Look Up Table と補間を使用してセル遅延を計算する。ここで、セル遅延とは、入力ピンの電圧が動作電源電圧の 50% になってから出力ピンの電圧が動作電源電圧の 50% になるまでの時間をいう (図 2)。セル遅延を算出するには、図 3 に示すような出力負荷容量と入力遷移時間をインデックスとして使用する 2 次元テーブル (以下、セル遅延テーブル) を用いて以下のように求める：

- 1) セル U1 の遅延テーブルから前段セル U0 の出力遷移時間と U1 の出力負荷容量をインデックスとして U1 の出力遷移時間とセル遅延時間を求める。
- 2) 1)で求めた U1 の出力遷移時間と U2 の出力負荷容量をインデックスとして U2 の出力遷移時間とセル遅延時間を求める。
- 3) 以降、同様の操作を繰り返す。

このとき、入力遷移時間と出力負荷容量をインデックスとしてセル遅延テーブルからセル遅延を求めるため、インデックス値を含む 4 つの隣接するテーブル値を選択し、4 点で構成される面の近似値を求めるためにガウスの消去法を用いて 2 次元補間による係数決定が行われる。

2.2 基本セルの遅延特性のモデル化

上記の入力遷移時間と出力負荷容量をインデックスとするセル遅延テーブルに対して、最小 2 乗法による多項式近似を行うことにより係数を最適化し、回路遅延特性を数式で表現することを考える。

基本セルの遅延特性として、ここではインバータセルの立下がり遅延について考える。セルの立下がり遅延テーブルは図 3 に示す出力遷移時間テーブル、遅延時間テーブルとも入力遷移時間と出力負荷容量をインデックスとする 7×7 のテーブルで与えられているとする。このとき、該テーブルを最小 2 乗法を用いて係数の最適化を行い、1 次から 6 次までの近似多項式で表した場合の項数と元の 2 次元テーブル値に対する RMS 誤差を図 4 に示す。また、図 5 は 1 次、3 次および 4 次の多項式で近似した場合の入力遷移時間と出力負荷容量に対するセルの立下がり遅延時間をプロットしたものである。近似多項式による数式表現を用いてモデル化を行うためには、より低次で高精度な多項式が必要である。しかし、図 4、図 5 からわかるように、高次になるとともに高精度になるが、4 次以上になると極が存在する。これを避けるため、3 次多項式での近似を採用することにする（図 6）。

3 1 ビット全加算回路の設計

3.1 総遅延の算出

実回路への適用例として、図 8 に示す 1 ビット全加算回路の設計を行う。設計値は各セル間を接続する配線長（図 8 の L1,L2,L3,L4,L5）、設計制約は入力 A から出力 S への遅延の上限/下限および各セルの出力負荷容量の上限とする。また、仕様値として、回路への入力信号の遷移時間（Input_slew）、および出力負荷容量（Output_load）が与えられているとする。ここでは、与えられた Input_slew、Output_load に対する各配線長の最適化を行う。なお、配線遅延モデルは図 7 に示す Elmore 遅延モデルを用いることにする[3][6]。

図 9 に Mathematica を用いて回路の入力 A から出力 S までのトータル遅延を求める計算フローを示す。2.2 節で述べたように各セルの遅延特性を入力遷移時間と出力負荷容量をパラメータとする 3 次の近似多項式でモデル化する。ここで、出力負荷容量は負荷配線の容量と次段セルの入力ピン容量の合計である。図 9 で、Dcellx, Tcellx は各々セル x の遅延、出力遷移時間を、DLYx, Tcellx はそれらの関数、R0, C0 は各々製造工程から得られる単位長さあたりの配線抵抗値 ($\Omega/\mu\text{m}$) と配線容量値 ($\text{F}/\mu\text{m}$) を表す。本計算フローで得られたトータル遅延を Input_slew、Output_load、および各配線長（Length1,...,Length5）をパラメータとする多項式により近似を行った結果、27 次の近似多項式が得られた。

3.2 配線長の最適化

上記 3.1 で求めたトータル遅延は Input_slew、Output_load と各配線長（Length1,...,Length5）をパラメータとする関数（TotalDLY）で表される。また、各セル毎の出力負荷容量は、配線負荷（単位長さあたりの配線容量 $C0(\text{F}/\mu\text{m})$ ）* 配線長に次段セルの入力ピン容量（Cpin）を加えたものである。制約条件として以下の不等式で表される条件が与えられている。

$$(1) \quad D_{\min} \leq \text{Total 遅延} \leq D_{\max}$$

$$(2) \quad \text{load1} \leq C_{\max1}, \quad \text{load2} \leq C_{\max2}, \quad \text{load3} \leq C_{\max3}, \quad \text{load4} \leq C_{\max4}, \quad \text{load5} \leq C_{\max5}$$

ただし、Dmin, Dmax は仕様値、load1, ..., load5 は Cell1 から Cell5 の出力負荷、Cmax1, ..., Cmax5 は製

造工程値から得られる制約条件である。

今、(1)(2)の制約条件の下、任意に与えた n 個の Input_slew と Output_load の組み合わせに対して配線長 ($\text{Length1}, \dots, \text{Length5}$) の最適化処理を実行する。その結果、図 10 に示すような Input_slew 、 Output_load と最適化された配線長のテーブルが得られる。この配線長テーブルから、 $\text{Length1}, \dots, \text{Length5}$ の各配線長を Input_slew 、 Output_load の 2 つをパラメータとする近似多項式の数式表現によりモデル化を行うため、3 次多項式で近似する (図 11)。得られた近似多項式を図 10 の配線長テーブル値と比較すると、RMS 誤差はすべて 1% 以内で近似できていることがわかる (図 12)。

4 まとめ

数式処理の応用として、数式処理ソフト **Mathematica** を用いて論理回路の遅延計算を行う方法を提案した。まず、基本セルの遅延特性を 3 次の多項式の数式表現によりモデル化した。モデル化の誤差は 4.2% である。さらに、本手法を用いて 1 ビット全加算回路を設計するため、当該回路の遅延特性を 27 次の多項式で近似し、配線長の最適化を 3 次の近似多項式で行った (誤差 1.0% 以内)。これらにより、数式処理による遅延計算に対する提案手法の有用性を確認することができた。

5 今後の課題

LSI 設計分野への数式処理の適用として、本提案手法の多ビット加算器や乗算器など本格的な回路設計への適用や、半導体基本素子 (トランジスタ、ダイオード、容量など) のモデル化、さらに、LSI レイアウトの配置・配線の最適化へと発展させることが今後の課題である。

参考文献

- [1] Sakurai, "Closed-form expressions for interconnection delay, coupling, and crosstalk in VLSI's," IEEE Trans. Electron Devices, vol.40, no.1, pp.118-124, Jan. 1983.
- [2] H.B.Bakoglu, "Circuits, Interconnections, and Packaging for VLSI," Addison-Wesley Publishing Company, 1990.
- [3] Chung-Kuan Cheng, "Interconnect Analysis and Synthesis," Wiley, 1999.
- [4] N.D.Arora, "Modeling and Characterization of Copper interconnects for VLSI Design," Nanotech2003.
- [5] Synopsys, Inc., LIBRARY COMPILER UserGuide, Vol2, V-1998.08,ed., 1998.
- [6] W.C.Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," *Journal of Applied Physics*, vol. 19, pp. 55-63, Jan. 1948.
- [7] Wolfram research, MATHEMATICA 5.2.

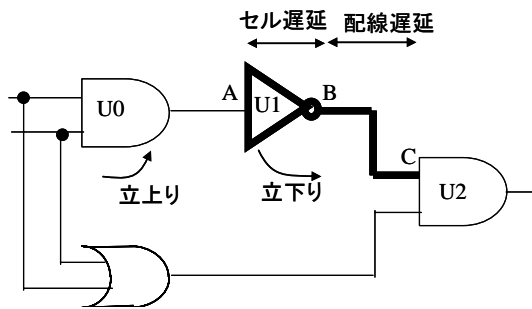


図 1 CMOS 非線形遅延モデルの構成要素

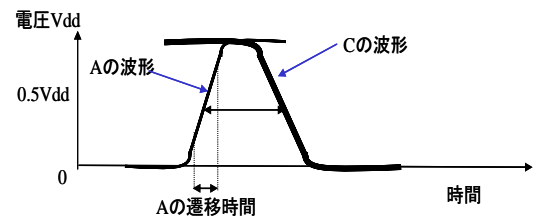


図 2 遅延の定義

セル遅延テーブル

出力負荷容量 入力遷移時間	C1	C2	C3	
T1	D11	D12	D13	
T2	D21	D22	D23	
T3	D31	D32	D33	
...				

遷移時間テーブル

出力負荷容量 入力遷移時間	C1	C2	C3	
T1	S11	S12	S13	
T2	S21	S22	S23	
T3	S31	S32	S33	
...				

図 3 遅延テーブル

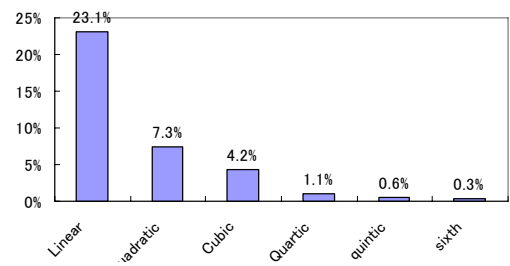
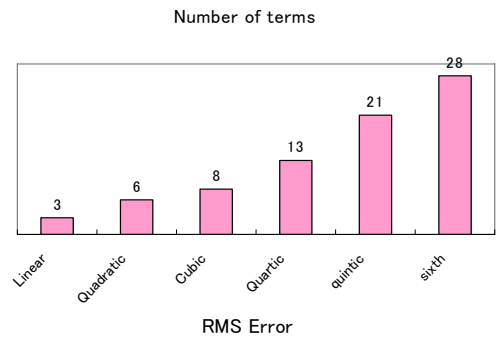


図 4 近似多項式の項数と RMS 誤差

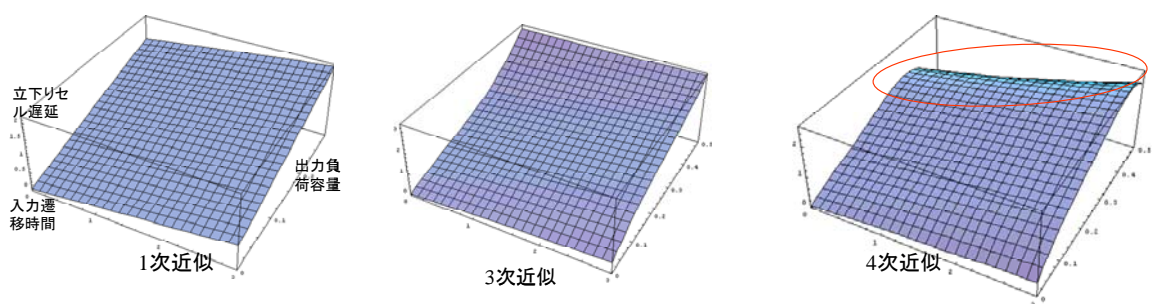
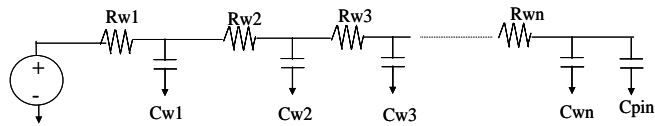


図 5 インバータセル遅延の近似例

$$\begin{aligned} \text{INV_fall}(x,y) = & -0.000986861 + 0.205682x - 0.0240294x^2 + 0.000884093x^3 + 5.7546y + \\ & 1.17031xy + 0.0900426x^2y - 11.6989y^2 - 3.32541xy^2 + 25.7925y^3 \end{aligned}$$

x: 入力遷移時間
y: 出力負荷容量

図 6 インバータセル遅延の近似式算出例



$$D_{\text{wire}} = R_{w1}C_{w1} + (R_{w1} + R_{w2}) \cdot C_{w2} + \dots + (R_{w1} + R_{w2} + \dots + R_{wn}) \cdot C_{wn}$$

特にすべての $R_j = R$, $C_j = C$ の時は

$$D_{\text{wire}} = RC \cdot n(n+1)/2$$

C_{pin} : 製造工程値。
 配線抵抗: 単位長あたりの抵抗 (R_0) * 配線長。 R_0 = シート抵抗/配線幅。
 配線容量: 単位長あたりの容量 (C_0) * 配線長。 C_0 は Sakurai 式による 1 次元モデル。
 ただし、配線幅一定。

図 7 配線遅延モデル

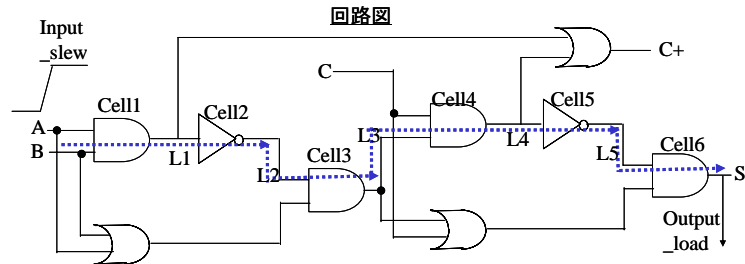


図 8 1 ビット全加算回路

Mathematicaによる実装例

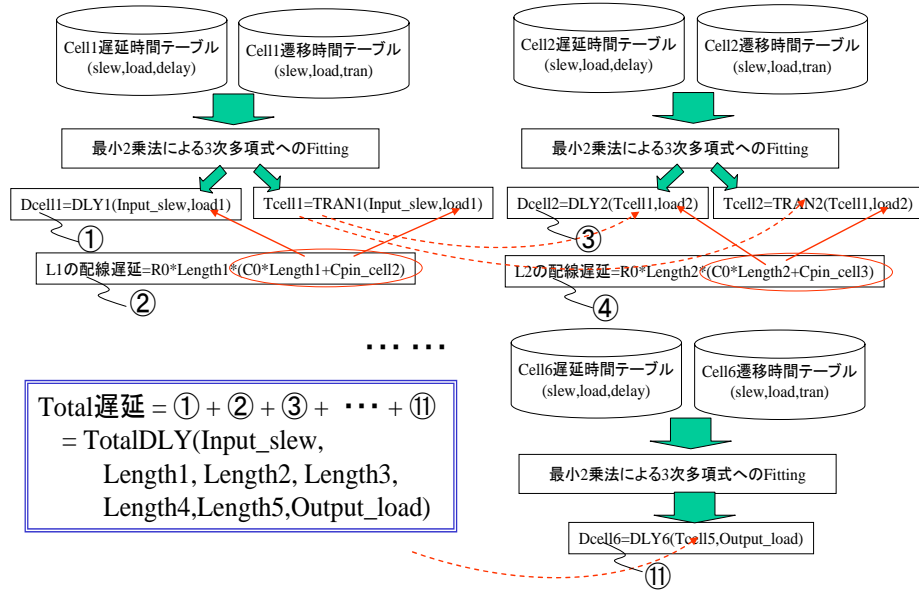


図 9 計算フロー

(Input_slew, Output_load, Length1, Length2, Length3, Length4, Length5)
 ($s_1, O_1, Lo_{1_1}, Lo_{2_1}, Lo_{3_1}, Lo_{4_1}, Lo_{5_1}$)
 ($s_2, O_2, Lo_{1_2}, Lo_{2_2}, Lo_{3_2}, Lo_{4_2}, Lo_{5_2}$)
 ($s_3, O_3, Lo_{1_3}, Lo_{2_3}, Lo_{3_3}, Lo_{4_3}, Lo_{5_3}$)
 ⋮
 ($s_n, O_n, Lo_{1_n}, Lo_{2_n}, Lo_{3_n}, Lo_{4_n}, Lo_{5_n}$)

図 10 配線長テーブル

$$\text{Length1} = 56.2706 + 14.0774 x - 25.6254 x^2 + 12.6111 x^3 + 352.036 y - 46.557 x y + 31.8844 x^2 y - 3466.32 y^2 + 54.5149 x y^2 + 5087.15 y^3$$

$$\text{Length2} = 166.617 + 65.6652 x - 152.015 x^2 + 93.0336 x^3 + 180.61 y + 706.275 x y - 344.757 x^2 y - 3095.31 y^2 - 1044.05 x y^2 + 6243.48 y^3$$

$$\text{Length3} = 137.207 + 67.3216 x - 204.595 x^2 + 135.224 x^3 - 1230.47 y + 424.636 x y - 205.38 x^2 y + 4804.82 y^2 - 514.248 x y^2 - 5743.89 y^3$$

$$\text{Length4} = 327.457 - 84.2874 x + 223.271 x^2 - 139.394 x^3 + 735.468 y - 901.97 x y + 419.273 x^2 y - 2882.1 y^2 + 1261.5 x y^2 + 2311.37 y^3$$

$$\text{Length5} = 119.946 + 78.9733 x - 240.862 x^2 + 152.843 x^3 - 2839.19 y + 1401.86 x y - 638.947 x^2 y + 17538.3 y^2 - 1942.6 x y^2 - 24435.2 y^3$$

図 11 配線長最適化結果

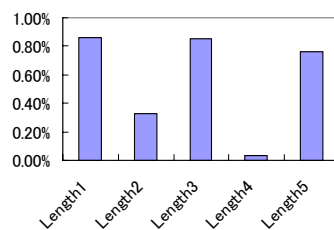


図 12 配線長テーブル値 vs.各近似式の RMS 誤差