J. Indian Inst. Sci., Sept.–Oct. 1987, 67, pp. 331–339.

Signal delay in RC networks*

A. A. Diwan†
Solid State Electronics Group, Tata Institute of Fundamental Research, Bombay 400 005, India.

Abstract

In this paper, a new method for approximating the step response of RC networks is presented. The method is useful in VLSI design, where it is important to estimate the delay introduced by an interconnection network. The method computes an N exponent approximation to the step response by approximating the Laplace transform of the response. An important feature of this approach is that the computation time depends only on the topology of the network and is independent of the values of the resistances and capacitances in the network. This is very useful when approximating the step response of stiff networks. Another feature of this approach is that distributed RC lines can also be simulated. The convergence of the approximations to the actual response as $N \to \infty$ is easily shown by using Laplace transforms.

Key words: RC networks, VLSI design, interconnection network, circuit simulation.

1. Introduction

Circuit simulation is one of the basic steps in the integrated circuit design process[1]. The increasing complexity of VLSI chips requires the development of faster methods for the simulation of these circuits. The conventional approach of formulating the network equations in the time domain and using standard numerical methods is no longer appropriate for circuits of this size. Several special methods, valid for particular classes of networks, have been developed[2].

One such class is that formed by RC networks. An RC network is a good model for an interconnection network in a VLSI circuit (fig 1). As device sizes become smaller, the delay introduced by such an interconnection network becomes comparable to the delay in the device itself. The estimation of this delay is therefore important in order to design timing fault-free circuits.

The delay is taken to be the rise time of the step response of the RC network. The approach that has been used previously is to compute the upper and lower bounds on the step response from which bounds on the delay can be computed[3–5]. The bounds on the step response are usually single exponential functions. In this paper, a different method is used to compute multi-exponent approximations to the step response. Although these approximations are not upper or lower bounds, their convergence to the exact response

*First presented at the Platinum Jubilee Conference on Systems and Signal Processing held at the Indian Institute of Science, Bangalore, India, during December 11–13, 1986.
†Present address: Department of Computer Science and Engineering, Indian Institute of Technology, Bombay 400 076.

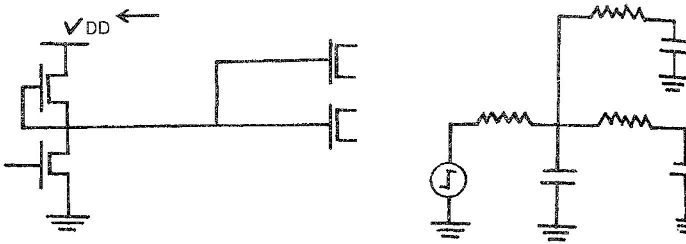

FIG. 1. MOS interconnect and its RC model.

implies that the delay is approximated directly, rather than computing bounds on the delay.

The paper is organised as follows. In section 2, the network equations are formulated in the transformed domain. In section 3, we present an algorithm for solving these equations and a complete example is given in section 4. Finally, we discuss the extension of this method to include networks with distributed elements, and illustrate it with an example.

2. Formulation of equations

Let $v(t) = [v_1(t), v_2(t), \ldots v_n(t)]^T$ be the vector of node voltages in the time domain. We assume that $v_1(t)$ is a step function. Let $V(s) = [V_1(s), V_2(s), \ldots, V_n(s)]^T$ be the vector of node voltages in the transform domain, $V_1(s) = 1/s$. We assume that there is no initial charge on the capacitors and also that there are no current sources in the network. The network equations can be formulated in the transform domain using the admittance matrix $Y(s)$ as

$$\sum_{j=1}^{n} Y_{ij}(s) V_j(s) = 0 \quad i = 2,3,\ldots,n, \tag{1}$$

or

$$\sum_{j=1}^{n} Y_{ij}(s)(sV_j(s)) = 0 \quad i = 2,3,\ldots, n.$$

The entries in the matrix $Y(s)$ are polynomials in s of degree at most 1, if the network contains only discrete resistances and capacitances. Let $V_j'(s) = sV_j(s)$, $V_1'(s) = 1$. Since $\lim s \to 0$ $V_j'(s)$ is the final value of $v_j(t)$, $V_j'(s)$ can be expanded in a power series as

$$V_j'(s) = \sum_{i=0}^{\infty} c_{ji} s^i . \tag{2}$$

Consider an N exponent approximation to $v_j(t)$ which satisfies the initial condition and denote it by $v_{ja}(t)$. Thus $v_{ja}(0) = 0, j = 2,3, \ldots, n$. Let $V_{ja}(s)$ be the Laplace transform of $v_{ja}(t)$. Then $sV_{ja}(s)$ is a rational function of s with numerator of degree $N-1$ and denominator of degree N. This is because lim $s \to \infty$ $sV_{ja}(s) = v_{ja}(0) = 0$, hence the numerator of $sV_{ja}(s)$ must be of smaller degree than the denominator. Again $V'_{ja}(s) = sV_{ja}(s)$ can be expanded in a power series as

$$V'_{ja}(s) = \sum_{i=0}^{\infty} d_{ji} s^i. \tag{3}$$

To compute the approximations, we require that $c_{ji} = d_{ji}$ for $i = 0,1,2,..,2N-1$. There are $2N$ unknowns in $V'_{ja}(s)$ which can be computed from the $2N$ values of d_{ji}, $i = 0,1,..,2N-1$. Therefore $V'_{ja}(s)$ is a $(N-1, N)$ Padé approximation to $V'_j(s)$. Hence to compute $V'_{ja}(s)$ we need to know only the coefficients up to the $2N-1$th power of s in the power series expansion of $V'_j(s)$.

Let $V_j^-(s)$ be the truncated power series $\sum_{i=0}^{2N-1} c_{ji} s^i$, a polynomial of degree $2N-1$.

Then $V_j^-(s)$ also satisfies the equation

$$\sum_{j=1}^{n} Y_{ij}(s) V_j^-(s) = 0 \quad i = 2,3, \ldots, n. \tag{4}$$

This follows from the fact that $Y_{ij}(s)$ is a polynomial of degree at most 1; therefore in the product $Y_{ij}(s)V'_j(s)$ in equation (1), the coefficient of s^i depends only on c_{ji} and $c_{j(i-1)}$ and does not depend on c_{jk} for $k > i$. Thus to compute $V_j^-(s)$ we can work with polynomials of degree $2N-1$ rather than infinite series. The remaining part of the series is irrelevant as far as computing $V_j^-(s)$ is concerned.

We next show that the above method gives a 'good' approximation. Let $e_j(t) = v_j(t) - v_{ja}(t)$ be the error function. Then $E_j(s) = V_j(s) - V_{ja}(s)$ and $E'_j(s) = V'_j(s) - V'_{ja}(s)$. Expanding $E'_j(s)$ in a power series, equations (2) and (3) and the conditions $c_{ji} = d_{ji}, i = 0, 1, \ldots, 2N-1$, imply that the coefficient of s^i is 0 for $i = 0, 1, \ldots, 2N-1$. This means that the coefficient of s^i in the power series expansion of $E_j(s)$ is 0 for $i = 0,1,\ldots,2N-2$. Therefore

$$d^i/ds^i \; E_j(s)|_{s=0} = 0, \; i = 0,1,2,\ldots,2N-2. \tag{5}$$

From elementary Laplace transform theory, we have

$$d^i/ds^i \; E_j(s) \,|_{\,s=0} = \int_0^\infty (-t)^i \; e_j(t) \; dt. \tag{6}$$

Equations (5) and (6) imply that the error function $e_j(t)$ satisfies

$$\int_0^{\infty} t^i e_j(t)\ dt = 0 \quad i = 0,1,\ldots,2N-2. \tag{7}$$

Hence as $N \to \infty$, equation (7) implies that the error function is orthogonal to all polynomials and hence is identically 0. For a given value of N, equation (7) implies that the error function is orthogonal to all polynomials of degree less than $2N-1$. This gives a quantitative measure of the accuracy of the approximation.

For a discrete network with k capacitors, when $N = k$, $v_j(t) = v_{ja}(t)$; however, computationally this is very expensive when k is large. This is usually the case, as in the RC models of interconnection networks where every node has a capacitance to ground. However, we can obtain good approximations by choosing N to be much smaller than the number of nodes. When the network contains distributed elements, no finite value of N can give the exact response, but equation (7) implies that the approximations converge to the exact response.

3. The algorithm

In this section we will briefly describe the algorithms for computing the approximations derived in Section 2. The basic equation to be solved is equation 4,

$$\sum_{j=1}^{n} Y_{ij}(s) V_j^-(s) = 0 \quad i = 2,3,\ldots,n \quad V_1^-(s) = 1.$$

This is a system of linear equations where the matrix elements and the unknowns are polynomials in s. The method used for solving it is the same as that used for solving ordinary linear equations, such as Gaussian elimination, except that the arithmetic is now performed on polynomials. Let $Y^-(s)$ be the inverse of the matrix $Y(s)$. The entries in $Y^-(s)$ are rational functions of s. However, since we are not interested in computing the actual inverse, but only in the truncated power series expansions of the elements of $Y^-(s)$, we do not have to work with rational functions. Therefore, when dividing one polynomial by another, we expand the quotient in a power series and retain terms only up to the $2N-1$th degree. We can always do this as long as the constant term in the denominator is non-zero. Otherwise pivoting may be used to get the appropriate denominator. As long as the resistance network is connected, the matrix is not singular. This is guaranteed in the RC models of interconnection networks.

The next step is to compute the Padé approximation $V'_{ja}(s)$ from the values of $V_j^-(s)$ computed in the first step[6]. This can be done by using standard methods, and it requires only the solution of a small system of linear equations. Finally to compute $v_{ja}(t)$ from $V'_{ja}(s)$, we expand $V'_{ja}(s)$ into partial fractions and use the standard methods of inverting Laplace transforms. This gives $v_{ja}(t)$ as a sum of N exponentials and a constant. This approximation enables us to estimate the delay as the rise time of the step response, which may be appropriately defined.

Usually $N=2$ gives a sufficiently good approximation to the step response. When $N=2$, the computations in step 2 are considerably simplified, and most of the computation time is taken in the first step. This time depends only on the topology of the network and not on the values of the elements. Thus if the network is a tree the computation time depends linearly on the number of nodes. This is in sharp contrast to the case when time domain methods are used to simulate RC networks. The stability criteria require that the step size be smaller than the smallest time constant in the network, whereas the response may be governed by a much larger time constant.

4. Example

In this section we illustrate the method with an example. Consider the network shown in fig. 2. This is a stiff circuit as the time constants differ by about two orders of magnitude.

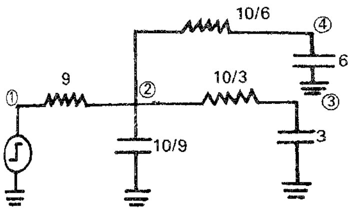

FIG. 2. A stiff RC network.

Such a circuit poses considerable difficulty when simulated by a conventional algorithm. The step size should be smaller than the smallest time constant, but the circuit has to be simulated for a long time depending on the largest time constant. This simple circuit required about 1.5 seconds when simulated by SPICE2[7] on a CDC machine. On the other hand, we show the simple computations performed by our algorithm with $N=2$, and compare the approximation with the actual response. In the transform domain the network equations can be formulated as

$$\begin{bmatrix} 91/90+10/9s & -3/10 & -6/10 \\ -3/10 & 3/10+3s & 0 \\ -6/10 & 0 & 6/10+6s \end{bmatrix} \begin{bmatrix} \tilde{V}_2(s) \\ \tilde{V}_3(s) \\ \tilde{V}_4(s) \end{bmatrix} = \begin{bmatrix} 1/9 \\ 0 \\ 0 \end{bmatrix}.$$

Divide the second row by $3/10+3s$ and the third row by $6/10+6s$ retaining terms up to s^3 to get

$$\begin{bmatrix} 91/90+10s/9 & -3/10 & -6/10 \\ -1+10s-100s^2+1000s^3 & 1 & 0 \\ -1+10s-100s^2+1000s^3 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{V}_2(s) \\ \tilde{V}_3(s) \\ \tilde{V}_4(s) \end{bmatrix} = \begin{bmatrix} 1/9 \\ 0 \\ 0 \end{bmatrix}.$$

Add 3/10 times of the second row and 6/10 times of the third row to the first row to get

$$\begin{bmatrix} 1/9+91s/9-90s^2+900s^3 & 0 & 0 \\ -1+10s-100s^2+1000s^3 & 1 & 0 \\ -1+10s-100s^2+1000s^3 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_2^-(s) \\ V_3^-(s) \\ V_4^-(s) \end{bmatrix} = \begin{bmatrix} 1/9 \\ 0 \\ 0 \end{bmatrix}$$

This gives $V_2^-(s) = 1/(1+91s-810s^2+8100s^3)$
$= 1-91s+9091s^2-909091s^3.$

$V_3^-(s) = V_4^-(s) = 1-101s+10101s^2-1010101s^3.$

Let $V_{2a}'(s) = a_0 + a_1 s/1 + b_0 s + b_1 s^2$ be a Padé approximation to $V_2^-(s)$.

Then

$$a_0 + a_1 s = (1-91s+9091s^2-909091s^3)(1+b_0 s+b_1 s^2).$$

Considering the coefficients of s^2 and s^3 on both sides, we get

$$9091 - 91b_0 + b_1 = 0$$

$$-909091 + 9091b_0 - 91b_1 = 0$$

which gives $b_0 = 101,\ b_1 = 100$

and $a_0 = 1,\ a_1 = 10.$

Therefore, $V_{2a}'(s) = \dfrac{1+10s}{1+101s+100s^2}$

and $V_{2a}(s) = \dfrac{(1+10s)}{s(1+101s+100s^2)}.$

Expanding in partial fractions

$$V_{2a}(s) = \frac{1}{s} - \frac{1}{11(s+1)} - \frac{10}{11(s+1/100)}$$

which gives $v_{2a}(t) = 1 - \dfrac{e^{-t}}{11} - \dfrac{10e^{-t/100}}{11}.$

A similar calculation gives

$$v_{3a}(t) = v_{4a}(t) = 1 + \frac{e^{-t}}{99} - \frac{100e^{-t/100}}{99}.$$

In this case, the actual response also has only two exponentials even though there are three capacitors. Hence the computed response is also the actual response. However, if we use an approximation with $N = 1$, we obtain

$$v_{2a}(t) = 1 - e^{-t/91}$$

and

$$v_{3a}(t) = 1 - e^{-t/101}.$$

If we define the delay to be the time required for the response to reach 90% of the final value, then for $v_2(t)$ we obtain

$$t_d = 210 \ (N = 1), \ t_d = 221 \text{ with } N = 2.$$

This shows that with $N = 1$, a good approximation to the step response is obtained.

4. Conclusions

We have described a method for approximating the step response of an RC network. This has been done for the special case when all the elements in the network are discrete and all the capacitors in the network are initially uncharged. However, these restrictions can be easily removed. An initial charge on a capacitor can be modelled by a constant current source in the transform domain. If the network contains distributed elements, the elements of the admittance matrix are no longer polynomials in s, but can be expanded in a power series. Again, by retaining terms up to the $2N-1$th power of s only, the first $2N-1$ coefficients in the power series expansion of the Laplace transform of the step response can be computed as above. In this case, no finite value of N will give the exact response, but the approximations will converge to the actual response as $N \to \infty$.

As an illustration, consider a uniformly distributed transmission line of length L, with resistance r and capacitance c per unit length (fig. 3).

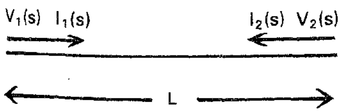

FIG. 3. A transmission line.

Assume that there is no initial charge on the line. The network equations in the transform domain for this line can be written as

$$\begin{bmatrix} I_1(s) \\ I_2(s) \end{bmatrix} = \begin{bmatrix} \sqrt{\frac{sc}{r}}\coth \sqrt{scr}\, L & -\sqrt{\frac{sc}{r}}\,\text{cosech}\, \sqrt{scr}\, L \\ -\sqrt{\frac{sc}{r}}\,\text{cosech}\, \sqrt{scr}\, L & \sqrt{\frac{sc}{r}}\coth \sqrt{scr}\, L \end{bmatrix} \begin{bmatrix} V_1(s) \\ V_2(s) \end{bmatrix}.$$

Suppose $V_1(s) = 1/s$ and the line is open circuited at the other end. Then $I_2(s) = 0$ which gives

$$V_2'(s) = sV_2(s) = \frac{1}{\cosh \sqrt{scr}\, L}.$$

Expanding this in a power series and retaining terms up to s^3, we get

$$V_2'(s) = 1 - \frac{scrL^2}{2} + \frac{5s^2c^2r^2L^4}{24} - \frac{61s^3c^3r^3L^6}{720}.$$

Assuming $crL^2 = 1$, then

$$V_2'(s) = 1 - \frac{s}{2} + \frac{5s^2}{24} - \frac{61s^3}{720}.$$

The Padé approximation to $V_2'(s)$ can be computed as before, and this gives

$$V_{2a}'(s) = \frac{120 - 4s}{120 + 56s + 3s^2}$$

$$V_{2a}(s) = \frac{120 - 4s}{s(120 + 56s + 3s^2)}.$$

Expanding $V_{2a}(s)$ in partial fractions

$$V_{2a}(s) = \frac{1}{s} - \frac{1.277}{s + 2.47} + \frac{0.277}{s + 16.2}.$$

Therefore

$$v_{2a}(t) = 1 - 1.277e^{-2.47t} + 0.277e^{-16.2t}.$$

Thus we obtain a better approximation to the step response than would be obtained by using discrete approximations to the distributed line.

References

1. CHUA, L.O. AND PEN-MIN, L. — *Computer aided analysis of electronic circuits*, Prentice-Hall, 1975.
2. NEWTON, A.R. — Techniques for the simulation of large-scale integrated circuits, *IEEE Trans.*, 1979, **CAS–26,** 741–749.
3. WYATT, J.L. JR. — Signal delay in RC mesh networks, *IEEE Trans.*, 1985, **CAS–32,** 507–509.
4. RUBINSTEIN, J., PENFIELD, P. JR. AND HOROWITZ, M.A. — Signal delay in RC tree networks, *IEEE Trans.*, 1983, **CAD–2,** 202–211.
5. LIN, T.M. AND MEAD, C.A. — Signal delay in general RC networks, *IEEE Trans.*, 1984, **CAD–3,** 331–349.
6. RALSTON, A. AND RABINOWITZ, P. — *A first course in numerical analysis*, McGraw-Hill, 1978.
7. NAGEL, L. — *SPICE2: A computer program to simulate semiconductor circuits*, ERL Memo UCB/ERL M75/520, Berkeley, CA, 1975.