$$\hat{\alpha}_j(n+1,k) = -\sum_{\nu=j+1}^{k} \alpha_\nu(n+1,k),$$
$$j = k-1,\ k-2, \cdots, 0$$

and similarly for BDF-2), step e),

$$\hat{\gamma}_j(n+1,k) = -\sum_{\nu=j+1}^{k+1} \gamma_\nu(n+1,k),$$
$$j = k, \cdots, 1.$$

A possible objection to the procedure described in this Appendix would be that numerical roundoff might make the numbers inaccurate after many updates of γ using (23). Therefore, it is recommended that at every ten steps, for example, one recompute γ using (21) directly, i.e., replace BDF-2), step d), by

$$\gamma_j(n+1,k) = F(n+1,k) \Big/ \left(A_{0j}{}^{n+1} \prod_{\substack{\nu=1 \\ \nu\neq j}}^{k+1} A_{j\nu}{}^{n+1} \right),$$
$$j = 2, \cdots, k+1.$$

References

[1] C. W. Gear, "The control of parameters in the automatic integration of ordinary differential equations," Dept. Comput. Sci., Univ. of Illinois, Int. Rep., May 1968; see also "The automatic integration of ordinary differential equations," *Comm. ACM*, vol. 14, Mar. 1971, pp. 176–179.

[2] W. E. Milne, *Numerical Calculus.* Princeton, N. J.: Princeton University Press, 1949.

[3] F. Odeh and W. Liniger, "A note on unconditional fixed-h stability of linear multistep formulae," to be published in *Computing*.

[4] A. Nordsieck, "On numerical integration of ordinary differential equations," *Math. Comp.*, vol. 16, 1962, pp. 22–49.

[5] P. J. Davis, *Interpolation and Approximation.* Waltham, Mass.: Blaisdell, 1963.

[6] S. Perlis, *Theory of Matrices.* Reading, Mass.: Addison-Wesley, 1953.

[7] G. D. Hatchtel, R. K. Brayton, and F. G. Gustavson, "The sparse tableau approach to network analysis and design," *IEEE Trans. Circuit Theory*, vol. CT-18, Jan. 1971, pp. 101–113.

[8] C. W. Gear, "The automatic integration of ordinary differential equations," in *Information Processing 68*, A. J. H. Morrell, Ed. Amsterdam, The Netherlands: North Holland, 1968, pp. 187–193.

[9] P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations.* New York: Wiley, 1962.

Steady-State Analysis of Nonlinear Circuits with Periodic Inputs

THOMAS J. APRILLE, JR., MEMBER, IEEE, AND TIMOTHY N. TRICK, MEMBER, IEEE

Abstract—In the computer-aided analysis of nonlinear circuits with periodic inputs and a stable periodic response the steady-state periodic response is found for a given initial state by simply integrating the system equations until the response becomes periodic. In lightly damped systems this integration could extend over many periods making the computation costly. In this paper a Newton algorithm is defined which converges to the steady-state response rapidly. The algorithm is applied to several nonlinear circuits. The results show a considerable reduction in the amount of time necessary to compute the steady-state response. In addition, the initial iterates give information on the transient response of the system.

Introduction

A VERY IMPORTANT problem in the computer-aided design of nonlinear circuits, yet unsolved, is the steady-state analysis of lightly damped nonlinear circuits, such as harmonic multipliers and oscillators. In these lightly damped systems even new stiff differential equation algorithms may require large amounts of computing time, since the transient response may be significant for a hundred cycles or more. This gives rise to expensive and perhaps inaccurate results. Several authors [1], [2] have attempted to solve this problem by harmonic balance techniques in which they assume that the response contains a fundamental component of a known frequency, plus several harmonics which they believe are dominant. This assumed solution is substituted into the differential equation and an optimization algorithm is used to adjust the amplitude and phase of the harmonics such that a mean-square-error function is minimized. This method has not been widely accepted for several reasons. First, the assumed solution may have no relation to the actual solution, e.g., the assumed solution may be unstable, or some important harmonic components might have been neglected. The error function will probably give no indication of these difficulties. Second, optimization techniques tend to have serious convergence problems and computation becomes excessive, particularly in higher order systems. For example, suppose that the fundamental frequency plus two harmonics are dominant. This gives us six unknowns (amplitude and phase of each harmonic) for each state variable. If the system of equations contains five state variables then we have 30 variables which must be optimized,

Manuscript received May 2, 1971; revised October 4, 1971. This work was supported by the Joint Services Electronics Program (U. S. Army, U. S. Navy, and U. S. Air Force), under Contract DAAB-07-67-C-0199.

The authors are with the Coordinated Science Laboratory, University of Illinois, Urbana, Ill. 61801.

and this is not a very high order problem! Finally, the harmonic balance algorithm is only useful for steady-state analysis, whereas the algorithm presented herein uses any of the numerical methods for the transient analysis of circuits. It finds a portion of the transient response which is used to predict the steady-state response, or the complete response can be determined if so desired. The examples show that the computational time is reduced significantly with this algorithm if only a portion of the transient response is found in addition to the steady-state response.

The Problem

Consider the system of equations

$$\dot{x} = f(x, t) \tag{1}$$

where x and f are n vectors; f is periodic in t of period T, is continuous in t, x has a continuous first partial derivative with respect to x for all x and $-\infty < t < \infty$. Henceforth, we will assume that (1) has a periodic solution $w(t)$ of period T.

Our goal is to determine the periodic state $w(0)$ such that integrating (1) from the initial state $w(0)$ over the interval $[0, T]$ we obtain the periodic solution $w(t)$ of period T. This is essentially a two-point boundary value problem in which the solution to (1) in the interval $[0, T]$ must satisfy the boundary condition.

$$x(0) = x(T). \tag{2}$$

Since

$$x(T) = \int_0^T f(x, \tau)d\tau + x(0) \tag{3}$$

we can express the above problem in terms of the mapping

$$x_0 = T(x_0) \tag{4a}$$

where

$$x_0 = x(0) \qquad T(x_0) = \int_0^T f(x, \tau)d\tau + x_0 \tag{4b}$$

and $x(t)$ satisfies (1) for $0 \le t \le T$.

One approach to the solution of (4) is the contraction mapping technique, i.e.,

$$x_0^{i+1} = T(x_0^i). \tag{5}$$

However, since $f(x, t)$ is periodic in t with period T, then

$$x_0^{i+1} = x[(i+1)T] = \int_0^{(i+1)T} f(x, \tau)d\tau + x_0^0. \tag{6}$$

Note that this says that we simply integrate (1) for a sufficient number of periods until the transient response becomes negligible. This is precisely the method currently used with the computer to arrive at the steady-state response. It is this method that we are seeking to improve as the convergence can be expensively slow, particularly in lightly damped systems. For example, consider the case when (1) is linear, i.e.,

$$\dot{x} = A(t)x + g(t) \tag{7}$$

where $A(t)$ and $g(t)$ are periodic in t with period T. The state transition matrix for this system has the form $\Phi(t, 0) = P(t)e^{Qt}$ where $P(t) = P(t+T)$, and $P(0) = I_n$. One can show that

$$(x^{m+1} - x^m) = e^{QT}(x^m - x^{m-1}). \tag{8}$$

Hence the iterates converge to the periodic solution $w(t)$ provided that the eigenvalues of e^{QT} all have magnitude less than one which is equivalent to saying that the eigenvalues of Q have negative real parts or that the system is asymptotically stable. Note that if the eigenvalues of Q are close to the imaginary axis, but in the left half plane, then the convergence can be quite slow. These results can be extended to the nonlinear system (1). If (1) has an asymptotically stable periodic solution $w(t)$ of period T, then there exists a ball with radius δ such that if $\|x_0 - w(0)\| < \delta$, then (5) will converge to the periodic point $w(0)$.

In the next several sections a Newton algorithm is derived which is shown to be computationally feasible, and examples are given which illustrate its advantage over the contraction mapping approach.

A Newton Algorithm for Nonlinear Circuits

The Newton iteration of (4) is defined by the equation [3]–[5]

$$x_0^{i+1} = x_0^i - [I - T'(x_0^i)]^{-1}[x_0^i - T(x_0^i)] \tag{9}$$

which can be put into the form

$$x_0^{i+1} = [I - T'(x_0^i)]^{-1}[T(x_0^i) - T'(x_0^i)x_0^i]. \tag{10}$$

Given the initial state x_0^i and the system of (1) a unique trajectory $x^i(t)$, $0 \le t \le T$, exists which is a solution to (1). Recall that $x^i(T) = T(x_0^i)$, therefore,

$$T'(x_0^i) = \left.\frac{\partial x(T, x_0)}{\partial x_0}\right|_{x_0^i} \tag{11}$$

But

$$\begin{aligned}\frac{\partial x(T, x_0)}{\partial x_0} &= \frac{\partial}{\partial x_0}\left[\int_0^T f(x(\tau), \tau)d\tau + x_0\right] \\ &= I + \int_0^T \frac{\partial f}{\partial x}\cdot\frac{\partial x(\tau)}{\partial x_0}d\tau \\ &= I + \int_0^T \frac{\partial f}{\partial x}d\tau + \int_0^T \frac{\partial f(\tau)}{\partial x} \\ &\quad \left[\int_0^\tau \frac{\partial f(\lambda)}{\partial x}\frac{\partial x(\lambda)}{\partial x_0}d\lambda\right]d\tau.\end{aligned} \tag{12}$$

This expansion can be continued and the resulting series defines a square matrix called a matrizant [6]. It represents the state transition matrix for the time-varying system

$$\dot{z} = \left.\frac{\partial f}{\partial x}\right|_{x^i(t)} z \tag{13}$$

where

$$\left.\frac{\partial f}{\partial x}\right|_{x^i(t)}$$

is the Jacobian of f evaluated along the trajectory $x^i(t)$, $0 \le t \le T$. Thus we denote

$$T'(x_0^i) = \Phi(T, 0; x_0^i) \tag{14}$$

where $\Phi(T, 0; x_0^i)$ is the state transition matrix of (13).

The above derivation can also be interpreted as follows. The desired periodic response satisfies the equations

$$w(t) = \int_0^t f(w(\tau), \tau)d\tau + w(0), \qquad 0 \le t \le T \tag{15a}$$

$$w(T) = w(0). \tag{15b}$$

Let us expand $f(w(t), t)$ in a Taylor series about the trajectory $x^i(t; x_0^i)$

$$f(w(t), t) = f(x^i(t), t) + \left.\frac{\partial f}{\partial w}\right|_{w = x^i(t)} (w(t) - x^i(t)) + R_2(w) \tag{16}$$

where $R_2(w)$ contains the higher order terms which are nonlinear. Let us drop these terms and define

$$y^i(t) = \int_0^t f(x^i(\tau), \tau)d\tau + \int_0^t F(\tau; x^i)(y^i(\tau) - x^i(\tau))d\tau + y^i(0), \quad 0 \le t \le T \tag{17}$$

where $F(\tau; x^i)$ is the Jacobian of f evaluated at $x^i(t)$ and $y^i(0)$ is yet to be defined. We can write (17) as

$$z^i(t) = \int_0^t F(\tau; x^i)z^i(\tau)d\tau + z^i(0), \qquad 0 \le t \le T \tag{18}$$

where $z^i(t) = y^i(t) - x^i(t)$. Note that $z^i(t)$ is the solution of the differential equation

$$\dot{z}^i = F(t; x^i)z^i. \tag{19}$$

The solution to (19) is

$$z^i(t) = \Phi(t, 0; x_0^i)z^i(0) \tag{20}$$

where $\Phi(t, 0; x_0^i)$ is the state transition matrix for (19). Now set $y^i(T) = y^i(0)$ and if $[I - \Phi(T, 0; x_0^i)]^{-1}$ exists, then

$$y^i(0) = [I - \Phi(T, 0; x_0^i)]^{-1} \cdot [x^i(T) - \Phi(T, 0; x_0^i)x_0^i]$$

or

$$y^i(0) = [\Phi^{-1}(T, 0; x_0^i) - I]^{-1} \cdot [\Phi(T, 0; x_0^i)x^i(T) - x_0^i]. \tag{21}$$

The above equations are equivalent to (9) and (10).

The reader should note at this point that in the linear case defined previously

$$\dot{x} = A(t)x + g(t) \tag{7}$$

the periodic solution satisfies the equations

$$w(t) = \Phi(t, 0)w(0) + \int_0^t \Phi(t, 0)\Phi^{-1}(\tau, 0)g(\tau)d\tau \tag{22a}$$

$$w(T) = w(0) \tag{22b}$$

where the constraint $w(T) = w(0)$ yields

$$w(0) = [I - \Phi(T, 0)]^{-1}[x(T) - \Phi(T, 0)x_0]. \tag{23}$$

Thus if the matrix $[I - \Phi(T, 0)]$ is nonsingular, that is, the equation

$$\dot{x} = A(t)x \tag{24}$$

does not possess a periodic solution with period T except for the trivial solution $x = 0$, then there exists a periodic solution of period T to (7) whose periodic point is given by (23). It is obvious that the Newton iteration converges in one iterate in this case as one would expect, whereas the contraction mapping technique is highly dependent on the characteristic multipliers of the state transition matrix.

The steady-state analysis algorithm based upon Newton's method can now be summarized.

Step 1: For the given initial state x_0^i compute the solution $x^i(t; x_0^i)$, $0 \le t \le T$, of (1).

Step 2: Compute the state transition matrix $\Phi(T, 0; x_0^i)$ of (19).

Step 3: Let $x_0^{i+1} = y_0^i$ which is obtained from (21).

Step 4: Return to step 1 with x_0^{i+1} unless $\|x^i(T) - x_0^i\| < \epsilon$ and $\|x_0^{i+1} - x_0^i\| < \delta$, where ϵ and δ are arbitrarily small positive numbers.

Step 5: Stop.

The convergence of this algorithm is discussed in Appendix I.

In the following sections the numerical calculations of $x^i(T)$ and $\Phi(T, 0; x_0^i)$ are discussed and some examples are given.

Numerical Computations

Electronic circuit analysis problems generally have widely separated eigenvalues and hence are said to be stiff. Recent research [8]–[11] has shown that implicit integration methods are the most efficient multistep methods for obtaining the solution of a system of stiff ordinary differential equations. If the function f is nonlinear, the application of these implicit integration methods results in a set of nonlinear algebraic equations which must be solved at each step. It has been shown that Newton's method is the most effective way to solve these equations. Hence the Jacobian of f must be determined at each step. In this section we will show that with the above method the state transition matrix $\Phi(T, 0; x_0^i)$ can be easily computed as it is related to the Jacobian of f along the trajectory of x^i. For simplicity the backward Euler formula with a fixed step size will be used to demonstrate a numerical method for approximating $x(T)$ and $\Phi(T, 0; x_0)$ where the superscript i, which denotes the iteration number, has been deleted to avoid confusion with the iterates which will be necessary to solve the implicit equations. The following results can easily be extended to higher order methods with variable step size.

Applying the backward Euler formula to (1) we obtain

$$\hat{x}(jh) = \hat{x}((j-1)h) + hf(\hat{x}(jh), jh) \tag{25}$$

where $h = T/k$ is the step size and $j = 1, 2, \cdots, k$. Equation (25) is solved for $\hat{x}(jh)$ by Newton's method

$$\hat{x}^{i+1}(jh) = \hat{x}^i(jh) - [I - hF(\hat{x}^i(jh))]^{-1} \cdot [\hat{x}^i(jh) - \hat{x}((j-1)h) - hf(\hat{x}^i(jh))] \tag{26}$$

where $F(x)$ is the Jacobian of f and $\hat{x}(kh) \approx x(T)$.

If we apply the backward Euler formula to (19), then

TABLE I
POWER SUPPLY EXAMPLE

Newton Algorithm				Contraction Mapping			
ith Iterate of Steady-State Algorithm	Computation Time to Obtain $x^i(0)$ and $x^{i-1}(T)$ (s)	$\lvert x_2^{i-1}(T)-w_2(0)\rvert$ where $w_2(0)=9.0555$	$\lvert x_3^{i-1}(T)-w_3(0)\rvert$ where $w_3(0)=0.0090285$	nth Period from $x^0(0)$	Computation Time to Obtain $x(nT)$ (s)	$\lvert x_2(nT)-w_2(0)\rvert$ where $w_2(0)=9.0555$	$\lvert x_3(nT)-w_3(0)\rvert$ where $w_3(0)=0.0090285$
1	59	7.1034	0.35803	2	54	5.7045	0.010214
2	88	1.6497	0.049609	4	88	2.8450	0.13704
3	114	0.2531	0.0064540	6	120	0.5141	0.0535873
4	140	0.0155	0.0003689	7	136	1.5532	0.0341340
5	165	0.0001	0.0000018	9	167	0.4584	0.0604030
6	190	0.	0.	10	182	0.7791	0.0005393
				25	408	0.0008	0.0107812
				50	781	0.0075	0.0004909
				75	1154	0.0008	0.0000312

$$\hat{z}(h) = \hat{z}(0) + hF(\hat{x}(h))\hat{z}(h)$$
$$\hat{z}(h) = [I - hF(\hat{x}(h))]^{-1}\hat{z}(0) \qquad (27)$$

and

$$z(T) \approx \hat{z}(kh) = \prod_{i=1}^{k} [I - hF(\hat{x}((k-i+1)h))]^{-1} z(0). \qquad (28)$$

Again we have dropped the subscript which denoted the number of the iterate. Thus

$$\Phi(T, 0; x_0) \approx \prod_{i=1}^{k} [I - hF(\hat{x}((k-i+1)h))]^{-1} \qquad (29)$$

or

$$\Phi^{-1}(T, 0; x_0) \approx \prod_{i=1}^{k} [I - hF(\hat{x}(ih))] \qquad (30)$$

where $\hat{x}(ih)$ was computed above. Thus the algorithm is very simple to implement.

Several examples follow which illustrate the power of this method in computing the steady-state response of weakly damped systems.

EXAMPLES

The first example is an equivalent power supply circuit given in Fig. 1. The state equations for this circuit are

$$\dot{x}_1 = \frac{1}{10^{-6}} \left\{ \tfrac{1}{5}[-x_1 - x_2 + 10\sin 60(2\pi)t] - i_d \right\}$$
$$\dot{x}_2 = \frac{1}{10^{-3}} \left\{ \tfrac{1}{5}[-x_1 - x_2 + 10\sin 60(2\pi)t] - x_3 \right\}$$
$$\dot{x}_3 = \frac{1}{0.1}[x_2 - x_4]$$
$$\dot{x}_4 = \frac{1}{10^{-3}} \left[x_3 - \frac{x_4}{1000} \right]$$
$$i_d = 10^{-6}[e^{40x_1} - 1].$$

The steady-state solution, for this circuit, was obtained by two methods, one by contraction mapping, and the other by use of

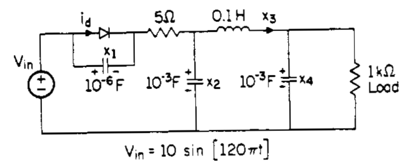

Fig. 1. Dc power supply circuit.

the steady-state Newton algorithm described in this paper. The numerical integration part of both methods used a variable step size prediction with Adam's second-order corrector. The corrector was solved by Newton's method. All computations were performed on a Control Data 1604 digital computer.

Both methods were started with the initial state vector $x(0)=0$. The period T was set equal to the forcing functions, i.e., $T=1/60$ s. The final value of the periodic point by both methods was $w_1(0)=-9.0743$, $w_2(0)=9.0555$, $w_3(0)=0.0090285$, and $w_4(0)=-9.1015$. The results of the two computer runs, for the variables x_2 and x_3 are summarized in Table I.

A second example, shown in Fig. 2(a), was constructed and analyzed in the same manner as the first circuit. The circuit shown in Fig. 2(a) is a frequency doubler; the 21-MHz input is doubled to a 42-MHz output by the nonlinearity of the transistor. For analysis purposes the transistor was modeled by the circuit shown in Fig. 2(b).

The initial starting point for both methods was $x_i^0=0.00001$, $i=1, 2, \cdots, 13$, with $T=1/21\times 16^6$. The final value obtained for the periodic point $w(0)$ was: $w_1(0)=6.50599$, $w_2(0)=-0.0211365$, $w_3(0)=19.5057$, $w_4(0)=21.7147$, $w_5(0)=-0.00235698$, $w_6(0)=-0.0308890$, $w_7(0)=-0.000561457$, $w_8(0)=0.00848798$, $w_9(0)=0.0874810$, $w_{10}(0)=-0.0653297$, $w_{11}(0)=11.7904$, $w_{12}(0)=11.9352$, and $w_{13}(0)=-0.00948132$.

The results of the two computer runs for this second example are summarized in Fig. 3. In Fig. 3 the norm of the error $\|x(nT)-w(0)\|$ for the contraction mapping method and the norm of the error $\|x^{i-1}(T)-w(0)\|$ for the steady-state Newton algorithm are plotted versus the computer execution time. The Euclidean norm was used. The figure indicates a considerable savings in computer time in using the steady-state Newton algorithm.

It should be noted that the results of the two examples were obtained on a relatively slow computer and no attempt was made to use the most efficient numerical methods. Thus the times given in Table I and Fig. 3 can be greatly improved, however the ratio

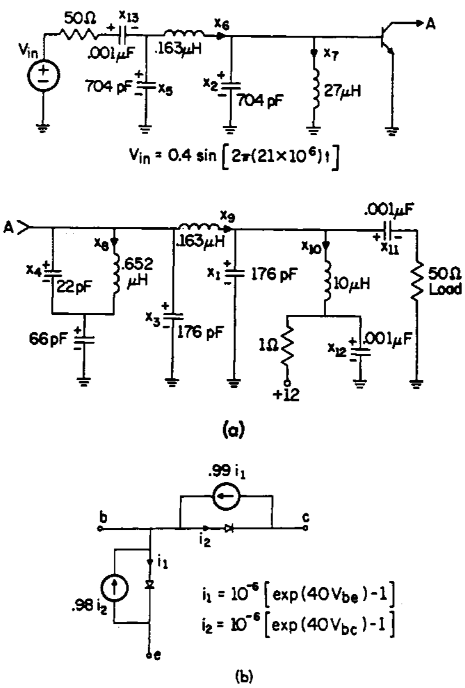

Fig. 2. (a) Frequency doubler circuit. (b) Transistor model.

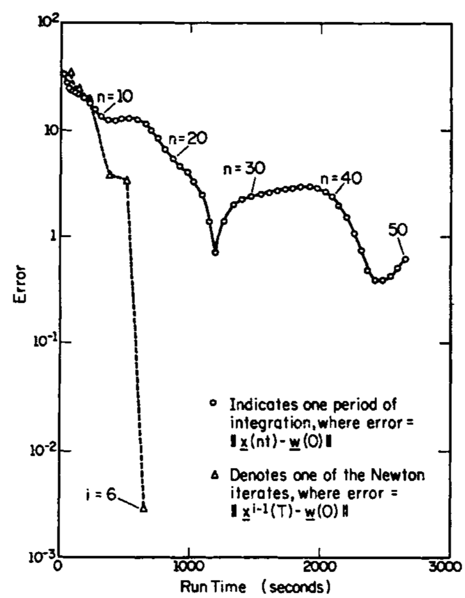

Fig. 3. Comparison between the contraction mapping and the Newton algorithm for finding the periodic point of the frequency doubler circuit.

of contraction mapping time to steady-state Newton algorithm time is problem dependent and should remain relatively constant for a given problem.

DC Analysis

It is interesting to note that the steady-state algorithm has all the necessary machinery to perform dc analysis. To perform dc analysis the period T is reduced to such a value as to allow the algorithm to numerically integrate only one step, and all input signals are assigned constant values.

This can be seen to yield the dc solution since by (9), $T(x_0^i)=x^i(h, x_0^i)$. If the numerical integration method is backward Euler (i.e., (25)) then this implies that $f(x^{i+1}(h))=0$, which yields the dc state $w(0)$.

A dc analysis was performed upon the circuit in Fig. 2(a) with $v_{in}=100$ and the initial conditions as follows. $x_1(0)=0.00001$, $x_2(0)=0.5$, $x_3(0)=0.05$, $x_4(0)=-0.6$, $x_5(0)=-1$, $x_6(0)=0.5$, $x_7(0)=0.5$, $x_8(0)=0.3$, $x_9(0)=-0.9$, $x_{10}(0)=0.3$, $x_{11}(0)=0.5$, $x_{12}(0)=0.5$, $x_{13}(0)=0.5$. The algorithm converged in 3 iterations to the following two place rounded values. $x_1(0)=12$, $x_2(0)=56\times10^{-14}$, $x_3(0)=12$, $x_4(0)=16\times10^{-14}$, $x_5(0)=12\times10^{-13}$, $x_6(0)=50\times10^{-9}$, $x_7(0)=70\times10^{-9}$, $x_8(0)=35\times10^{-11}$, $x_9(0)=-96\times10^{-8}$, $x_{10}(0)=-95\times10^{-8}$, $x_{11}(0)=12$, $x_{12}(0)=12$, $x_{13}(0)=100$.

If the iterates do not converge one can always integrate (1) for several steps of length h (h may be variable) and then apply the Newton algorithm again. If the bias point is globally stable, this process will eventually yield the solution.

Convergence to an Unstable Periodic Point

The Newton algorithm presented here for determining the steady-state response of dynamic systems can converge to an unstable periodic solution just as the Newton algorithm applied to the dc bias problem can converge to a singular point which is unstable. A discussion on convergence is given in Appendix I; however, the following example illustrates this point quite effectively.

Consider the Duffing's equation

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -0.2x_2 - x_1^3 + 0.3\cos t. \end{aligned} \tag{31}$$

Hayashi [17] has shown that this equation has three different periodic solutions, and hence, three different periodic states. Two are stable and one is unstable. The Newton algorithm was used to locate all three of these periodic states and solutions. With $x_1^0(0)=-0.382$ and $x_2^0(0)=1.45$ the Newton algorithm converged in three iterations to $w_1(0)=-0.3105931$ and $w_2(0)=0.0688257$ which is a stable periodic state. With $x_1^0(0)=0.027$ and $x_2^0(0)=1.1$ convergence occurred in five iterations to $w_1(0)=0.6263873$ and $w_2(0)=1.03347995$, the other stable periodic point. With $x_1^0(0)=-0.742$ and $x_2^0(0)=0.729$ convergence occurred in four iterations to $w_1(0)=-0.71598261$ and $w_2(0)=0.74740203$. This periodic state lies on the trajectory of the unstable periodic solution, because when the state equations where integrated from this periodic state convergence eventually occurred to the first stable steady-state trajectory with $w_1(0)=-0.3105931$ and $w_2(0)=0.0688257$.

Conclusions

A Newton algorithm has been presented for the computation of the steady-state periodic response of asymptotically stable systems. The algorithm has been shown to be very effective in reducing the amount of computation necessary for the analysis of nonlinear circuits. Of course, it is necessary to know that the circuit does have a periodic response with period T.

In the design of frequency multipliers, power supplies, and other nonlinear circuits which involve a periodic input of period T and a periodic response of period T the engineer can reasonably anticipate the Newton algorithm to yield the steady-state response in significantly less time than the usual numerical integration, particularly if the system is lightly damped. The reader should be aware of some of the properties of the convergence of this method which are analogous to the dc bias problem. For example if the periodic point is asymptotically stable and the initial state is sufficiently close to this periodic state the iterates will converge.

However, the method can converge to a periodic state which is not asymptotically stable just as the Newton iterates can converge to an unstable bias point. If the iterates do not converge, it can mean that a solution of period T does not exist, or that the initial iterate is not close enough to the periodic state. In these cases it is usually best to integrate the state equations for several periods, which is very easy to do since the necessary machinery already exists in the algorithm. During this integration various tests can be performed upon $x(nT)$ to determine if it would make an improved initial iterate for the steady-state algorithm. In fact, it has been noted that the Newton algorithm converges much more rapidly if one initially integrates for two periods, instead of one, before predicting the periodic state.

Finally, this algorithm is not useful for the oscillator problem. First, one does not usually know the exact period T of the oscillation. Second, even if the period T is known, as the periodic solution is approached one of the eigenvalues of the fundamental matrix of the equation of first variation tends to unity [7]. Thus the matrix $[I-\Phi(T, 0; \boldsymbol{w}_0)]$ is singular and cannot be inverted. An algorithm has been developed to overcome these problems and will be published shortly.

Appendix I

Convergence of Newton Iterates

Consider the differential equation

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, t) \tag{32}$$

where $\boldsymbol{x}$ and f are n vectors, and assume that

1) $f(\boldsymbol{x}, t+T)=f(\boldsymbol{x}, t)$ for all $\boldsymbol{x}, t$; $T>0$,

2) $f(\boldsymbol{x}, t)$ is continuous in t for all $(\boldsymbol{x}, t)\in R=\{\|x\|<\infty, t_0\le t\le t_0+T\}$,

3) $\partial f/\partial \boldsymbol{x}$ is continuous and uniformly bounded by $M<\infty$ for all $(\boldsymbol{x}, t)\in R$,

4) there exists a periodic solution $\boldsymbol{w}(t)$ to (32) of period T which satisfies the Liapunov asymptotic stability condition, i.e., the largest magnitude of the eigenvalues of $\Phi(t_0+T, t_0; \boldsymbol{w}_0) =\beta<1$ [8].

Conditions 2) and 3) above imply that there exists a unique continuous solution $\boldsymbol{x}(t; \boldsymbol{x}_0, t_0)$ which satisfies (32) and is continuous in $\boldsymbol{x}_0$ [15]. Let us define

$$\left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}(t;\boldsymbol{x}_0,t_0)} = A(t, \boldsymbol{x}_0) \tag{33}$$

then from hypothesis 3) it follows that the $n\times n$ matrix $A(t, \boldsymbol{x}_0)$ is bounded and continuous in $\boldsymbol{x}_0$, thus the sum

$$\Phi(t_0+T, t_0; \boldsymbol{x}_0) = I + \int_{t_0}^{t_0+T} A(\lambda, \boldsymbol{x}_0)d\lambda + \int_{t_0}^{t_0+T} A(\lambda, \boldsymbol{x}_0)\int_{t_0}^{\lambda} A(\xi, \boldsymbol{x}_0)d\xi d\lambda + \cdots \tag{34}$$

is uniformly and absolutely convergent [7]. Since the partial sums of (34) are continuous in $\boldsymbol{x}_0$ and, as shown previously, converge uniformly to $\Phi(t_0+T, t_0; \boldsymbol{x}_0)$, this implies that $\Phi(t_0+T, t_0; \boldsymbol{x}_0)$ is continuous with respect to $\boldsymbol{x}_0$ [16].

From hypothesis 4) it follows that $[I-\Phi(t_0+T, t_0; \boldsymbol{w}_0)]^{-1}$ exists, therefore, define

$$\delta \triangleq \|[I-\Phi(t_0+T, t_0; \boldsymbol{w}_0)]^{-1}\|. \tag{35}$$

Since $\Phi(t_0+T, t_0; \boldsymbol{x}_0)$ is continuous in $\boldsymbol{x}_0$, then for each $\epsilon>0$ there exists an open ball S_ϵ with center $\boldsymbol{w}_0$ and radius r_ϵ such that

$$\|[I-\Phi(t_0+T, t_0; \boldsymbol{w}_0)] - [I-\Phi(t_0+T, t_0; \boldsymbol{x}_0)]\| < \epsilon \tag{36}$$

for all $\boldsymbol{x}_0\in S_\epsilon$. If ϵ is chosen such that $\epsilon\delta<1$, then from the perturbation lemma [13] $[I-\Phi(t_0+T, t_0; \boldsymbol{x}_0)]^{-1}$ exists for all $\boldsymbol{x}_0\in S_\epsilon$ and

$$\|[I-\Phi(t_0+T, t_0; \boldsymbol{x}_0)]^{-1}\| \le \frac{\delta}{1-\epsilon\delta} \le P < \infty. \tag{37}$$

Because of the smoothness conditions assumed on $f(\boldsymbol{x}, t)$ it follows that $[\boldsymbol{x}_0-T(\boldsymbol{x}_0)]=\int_{t_0}^{t_0+T} f(\boldsymbol{x}, \tau)d\tau$, $\boldsymbol{x}(t_0)=\boldsymbol{x}_0$ is Frechet differentiable [13] at the point $\boldsymbol{w}_0$, i.e., for each $\sigma>0$ there is a ball S_σ with center $\boldsymbol{w}_0$ and radius r_σ such that for all $\boldsymbol{x}_0\in S_\sigma$

$$\|[\boldsymbol{x}_0-T(\boldsymbol{x}_0)] - [\boldsymbol{w}_0-T(\boldsymbol{w}_0)] - [I-\Phi(t_0+T, t_0; \boldsymbol{w}_0)](\boldsymbol{x}_0-\boldsymbol{w}_0)\| \le \sigma\|\boldsymbol{x}_0-\boldsymbol{w}_0\|. \tag{38}$$

Next we will show that the Newton iterates converge for a point $\boldsymbol{x}_0$ sufficiently close to $\boldsymbol{w}_0$.

Theorem

Consider the system (32) with solution $\boldsymbol{w}(t)$, $\boldsymbol{w}(t_0+T)=\boldsymbol{w}(t_0)$. The point $\boldsymbol{w}_0=\boldsymbol{w}(t_0)$ is a solution of $\boldsymbol{x}_0=T(\boldsymbol{x}_0)$ and is a point of attraction of the mapping

$$\boldsymbol{x}_0 = G(\boldsymbol{x}_0) \tag{39}$$

where $G(\boldsymbol{x}_0)\triangleq \boldsymbol{x}_0-[I-T'(\boldsymbol{x}_0)]^{-1}[\boldsymbol{x}_0-T(\boldsymbol{x}_0)]$, i.e., there exists a ball S with center $\boldsymbol{w}_0$ and radius r such that for all $\boldsymbol{x}_0^i\in S$ the point $\boldsymbol{x}_0^{i+1}\in S$ where $\boldsymbol{x}_0^{i+1}=G(\boldsymbol{x}_0^i)$ and

$$\lim_{i\to\infty} \boldsymbol{x}_0^i = \boldsymbol{w}_0.$$

Proof: Choose σ, ϵ, and the ball S such that $P(\sigma+\epsilon)<1$, $\epsilon\delta<1$, and $S\subset S_\sigma$ and $S\subset S_\epsilon$. We can write

$$\|G(\boldsymbol{x}_0)-G(\boldsymbol{w}_0)\| = \|x_0 - [I-\Phi(t_0+T, t_0; \boldsymbol{x}_0)]^{-1}\cdot[\boldsymbol{x}_0-T(\boldsymbol{x}_0)] - \boldsymbol{w}_0\|. \tag{40}$$

Let $Q=[I-\Phi(t_0+T, t_0; \boldsymbol{x}_0)]^{-1}$ where $\|Q\|\le P$ for all $\boldsymbol{x}_0\in S$. The following equation is equivalent to (40)

$$\begin{aligned}\|G(\boldsymbol{x}_0)-G(\boldsymbol{w}_0)\| &= \|QQ^{-1}(\boldsymbol{x}_0-\boldsymbol{w}_0) - Q[\boldsymbol{x}_0-T(\boldsymbol{x}_0)] \\ &\quad + Q[I-\Phi(t_0+T, t_0; \boldsymbol{w}_0)](\boldsymbol{x}_0-\boldsymbol{w}_0) \\ &\quad - Q[I-\Phi(t_0+T, t_0; \boldsymbol{w}_0)](\boldsymbol{x}_0-\boldsymbol{w}_0) \\ &\quad + Q[\boldsymbol{w}_0-T(\boldsymbol{w}_0)]\|.\end{aligned} \tag{41}$$

Thus grouping the second, third, and fifth term together and likewise the first and fourth we obtain

$$\|G(\boldsymbol{x}_0)-G(\boldsymbol{w}_0)\| \le P(\sigma+\epsilon)\|\boldsymbol{x}_0-\boldsymbol{w}_0\| \tag{42}$$

where $P(\sigma+\epsilon)<1$ for all $\boldsymbol{x}_0\in S$. Since $P(\sigma+\epsilon)<1$ and

$$\|\boldsymbol{x}_0^{i+1}-\boldsymbol{w}_0\| = \|G(\boldsymbol{x}_0^i)-G(\boldsymbol{w}_0)\| \le P(\sigma+\epsilon)\|\boldsymbol{x}_0^i-\boldsymbol{w}_0\| \tag{43}$$

then $\boldsymbol{x}_0^i\in S$ implies that $\boldsymbol{x}_0^{i+1}\in S$, furthermore,

$$\lim_{i\to\infty} \boldsymbol{x}_0^i = \boldsymbol{w}_0.$$

This completes the proof of convergence in the neighborhood of w_0.

One should note that one of the assumptions was that the periodic point w_0 be asymptotically stable. This was a sufficient condition for the existence of the inverse of $[I-\Phi(t_0+T, t_0; x_0)]$. However, it is possible for the Newton algorithm to converge to a periodic point which is not asymptotically stable. All that is necessary is that the matrix $[I-\Phi(t_0+T, t_0; x_0)]$ be invertible at each step and that the initial state x_0^0 be sufficiently close to the unstable periodic point w_0. This is analogous to the dc circuit bias problem in which the Newton algorithm can converge to an operating point which is dynamically unstable. To determine the stability of the periodic point w_0, one can always perturb w_0 and integrate (32) for several periods or calculate the eigenvalues of the matrix $\Phi(t_0+T, t_0; w_0)$.

Appendix II

Effect of Truncation Errors on the Periodic Point

Let $w(t)$ be the true periodic solution to (1) with period T and define $w^c(t)$ as the computed value. For simplicity assume that the backward Euler formula is used to compute $w^c(t)$, then

$$\begin{aligned}\varepsilon_{i+1} &= w_{i+1} - w_{i+1}{}^c \\ &= \left[I - h\frac{\partial f}{\partial x}\bigg|_{w_{i+1}{}^c}\right]^{-1}[\varepsilon_i + T_i - R_i] \qquad (44)\end{aligned}$$

where T_i is the truncation error and R_i the roundoff error. Assume that $R_i=0$, and let $n=T/h$ an integer, then

$$\begin{aligned}\varepsilon_n = w(T) - w^c(T) = M_n \cdots M_1\varepsilon_0 + M_n \cdots M_1 T_0 \\ + M_n \cdots M_2 T_1 + \cdots + M_n T_{n-1} \qquad (45)\end{aligned}$$

where

$$M_i = \left[I - h\frac{\partial f}{\partial x}\bigg|_{w_i{}^c}\right]^{-1}$$

and $M_n \cdots M_1 \approx \Phi(t_0+T, t_0; w_0)$. Let $w^c(T)=w_0{}^c+\delta$, then

$$\begin{aligned}\varepsilon_0 = [I - \Phi(t_0 + T, t_0; w_0)]^{-1} \\ \cdot[\delta + M_n \cdots M_1 T_0 + \cdots + M_n T_{n-1}]. \qquad (46)\end{aligned}$$

If we assume that $M_i \approx I$ for $i=1, 2, \cdots, n$, then $\|\varepsilon_0\|$ is proportional to $\|[I-\Phi(t_0+T, t_0; w_0)]^{-1}\|$ and to $\|\delta\|$ and $\|T_i\|$. If the Newton algorithm converges δ is quite small, e.g., $\delta \approx w^c(0)\times 10^{-8}$ on the CDC 1604 which is a 48-bit machine. Also, the truncation error can be controlled and kept small in most integration algorithms. Therefore, to a large extent, the error is determined by the first factor and if one or more eigenvalues of the matrix $\Phi(t_0+T, t_0; w_0)$ are close to unity, the error could be quite large as the following example illustrates.

Consider the differential equation

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 - 10^{-5}x_2 + 5\times 10^{-5}\sin t. \qquad (47)\end{aligned}$$

This is the differential equation for a simple tuned circuit with a $Q=10^5$. The steady-state solution is $w_1(t)=-5\cos t$ and $w_2(t)=5\sin t$. Therefore, the periodic state is $w_1(0)=-5$ and $w_2(0)=0$. This periodic state was computed with the Newton algorithm; first, with the integration time step fixed at $2\pi\times 10^{-3}$ s, and second with the integration time step fixed at $2\pi\times 10^{-4}$ s. Since the system is linear the algorithm converged in one iterate. In the first case $w_1{}^c(0)=-3.499$ and $w_2{}^c(0)=-2.293$. While in the second case, with an order of magnitude smaller time step and approximately three orders of magnitude smaller local truncation error, the algorithm converged to $w_1{}^c(0)=-5.013$ and $w_2{}^c(0)=-0.01833$. The norm $\|[I-\Phi(t_0+T, t_0; w_0)]^{-1}\|\approx 3\times 10^4$ for this problem. When $\Phi(t_0+T, t_0; w_0)$ is the state transition matrix of a linear time-invariant system, it can be shown that $\|[I-\Phi]^{-1}\|$ is proportional to Q. Thus it is necessary to keep the product of the truncation error with the highest Q of the circuit small.

It should be noted that the contraction mapping method would arrive at the same erroneous result and would require computer time measured in hours instead of seconds. The problem is that the rate of decay of the transient response is inversely proportional to Q. In the above example this decay time amounts to 10^6 s. Note that the period is 2π s. Thus even though the truncation error over one period is small, it may exceed the rate of decay of the transient response which is proportional to $(1/2Q)e^{-t/2Q}$. When this happens neither the Newton algorithm nor the contraction mapping method can converge to the correct periodic state.

References

[1] E. M. Baily, "Steady state harmonic analysis of nonlinear network," Ph.D. dissertation, Stanford Univ., Stanford, Calif., 1968.

[2] J. C. Lindenlaub, "An approach for finding the sinusoidal steady state response of nonlinear systems," in *Proc. 7th Ann. Allerton Conf. Circuit and Syst. Theory*, pp. 323–329, 1969.

[3] L. B. Rall, *Computational Solution of Nonlinear Operator Equations*. New York: Wiley, 1969.

[4] P. L. Falb and J. L. DeJong, *Some Successive Approximation Methods in Control and Oscillation Theory*. New York: Academic Press, 1969.

[5] H. B. Keller, *Numerical Methods for Two-Point Boundary-Value Problem*. Waltham, Mass.: Blaisdell, 1968.

[6] P. M. DeRusso, R. J. Roy, and C. M. Close, *State Variables for Engineers*. New York: Wiley, 1967.

[7] J. K. Hale, *Oscillations in Nonlinear Systems*. New York: McGraw-Hill, 1963.

[8] C. W. Gear, "Numerical integration of stiff ordinary differential equations," Dep. Comput. Sci., Univ. of Illinois, Rep. 221, Jan. 20, 1967.

[9] I. W. Sandberg and H. Shichman, "Numerical integration of systems of stiff nonlinear differential equations," *Bell Syst. Tech. J.*, vol. 47, pp. 511–527, Apr. 1968.

[10] H. Shichman, "Integration system of a nonlinear network-analysis program," *IEEE Trans. Circuit Theory*, vol. CT-17, pp. 378–386, Aug. 1970.

[11] W. Liniger and R. Willoughby, "Efficient integration methods for stiff systems of ordinary differential equations," *SIAM J. Numer. Anal.*, vol. 7, pp. 47–66, Mar. 1970.

[12] L. S. Pontryagin, *Ordinary Differential Equations*. Reading, Mass.: Addison-Wesley, 1962.

[13] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*. New York: Academic Press, 1970.

[14] R. Ralston, *A First Course in Numerical Analysis*. New York: McGraw-Hill, 1965.

[15] E. A. Coddington and N. Levinson, *Theory of Ordinary Differential Equations*. New York: McGraw-Hill, 1955.

[16] J. M. Holmsted, *Real Variables*. New York: Appleton, 1956.

[17] C. Hayashi, *Nonlinear Oscillations in Physical Systems*. New York: McGraw-Hill, 1964.