IEEE Transactions on Power Apparatus and Systems, Vol. PAS-96, no. 1, January/February 1977

A CLASS OF NEW MULTISTEP INTEGRATION ALGORITHMS FOR THE COMPUTATION OF POWER SYSTEM DYNAMICAL RESPONSE

George Gross
Pacific Gas & Electric
San Francisco, California 94106

Arthur R. Bergen
Dept. of Electrical Engineering and Computer Sciences
and the Electronics Research Laboratory
University of California, Berkeley, California 94720

ABSTRACT

The development of a class of efficient numerical integration schemes for computing power system dynamic response is presented. These schemes are derived by making detailed use of the structural properties of the differential-algebraic system representation of the multimachine power system. The nonlinear differential-algebraic system is split into a nonstiff part with long time constants coupled to a stiff part with a sparse Jacobian matrix whose longest time constant is shorter than that of the first part. These two parts are linear in their respective states, i.e. the system is semilinear. With the nonstiff part removed, a smaller set of stiff equations with a smaller conditioning number than the original system is obtained. Consequently, longer stepsizes can be used so as to reduce the computation time. The proposed multistep integration schemes exploit the sparsity, stiffness and semilinearity properties. Numerical results indicate that these schemes operate with good accuracy at stepsizes as large as 100 times those necessary to ensure numerical stability for conventional schemes.

INTRODUCTION

We shall be concerned here with the problem of efficient numerical integration for simulating power system dynamical behavior. Power system simulation studies involve the solution of a large differential-algebraic system of the form

$$\dot{\underline{x}} = \underline{X}(\underline{x},\underline{y}) \tag{1a}$$

$$\underline{0} = \underline{Y}(\underline{x},\underline{y}) \tag{1b}$$

The differential equations describe the dynamics of the machines and their associated control systems; the algebraic system includes the network steady state equations and the machine albegraic equations. In the investigation of system response to a sequence of events (for example those following a fault) the structure of (1) will change. In this case one investigates a sequence of initial value problems.

In practice, an important feature of the nonlinear differential system (1a) is its wide range of time constants, i.e. the eigenvalues of the Jacobian $\frac{\partial \underline{X}}{\partial \underline{x}}$ evaluated with $\underline{y} = \underline{y}(\underline{x})$ are widely spread. Such systems are termed *stiff*. Typically, the ratio of the largest to the smallest time constant may be of the order 10^3 or 10^4. The longest time constant then determines the length of the interval over which the system must be studied. On the other hand, to avoid numerical instability, integration methods must pick an integration stepsize sufficiently small to accommodate the shortest time constant, even though for most of this interval the components associated with this stepsize are negligible. Not only is this expensive, but, in addition, it may cause difficulty with roundoff error. One way to reduce the number of steps is to use multistep rather than Runge-Kutta methods ([15], p. 103). As yet, these multistep methods have not been widely adopted by the power industry and only a small number of investigations concerning their application have been reported in the literature. However, some well known multistep integration routines such as Milne, Hamming, Ralston and higher order Adams, have been examined and applied to small power systems [2], [4], [6], [7], [13]. In certain cases, stepsizes ten times as long as for Runge-Kutta methods were possible, although there were limits beyond which the stepsize could not be increased. This is because these methods, like those of the Runge-Kutta type, suffer from the shortest time constant barrier, i.e. to be numerically stable the stepsizes are controlled by the smallest time constant. This is true even for modified integration methods [6]. Among the Adams methods only the first (backward Euler) and second (trapezoidal rule) order Adams-Moulton are not restricted in this manner ([15], chapters 7 and 11); the choice of a stepsize for these integrations schemes is then restricted only by accuracy (the maximum allowable local truncation error) and not by stability. The trapezoidal rule with a fixed stepsize reduced the computation time of the transient stability simulation without sacrificing accuracy [9]. Results of a similar nature were reported using variable stepsize and variable order implicit integration schemes [11]. Oswald and Smith [8] by making efficient use of the linearity of part of the differential equations describing the power system dynamic behavior, developed a method that permits stepsizes a hundred times larger than Runge-Kutta methods when applied to a small power system. However, the evaluation of a certain high dimensional state transition matrix is required and this may discourage its application to large power systems (discussion of [8]). The use of a high dimensional state transition matrix to compute dynamical response is also called for in the solution method of [14]. In this paper we have obtained significant savings without the need to compute such state transition matrices. This has been accomplished by exploiting the structural properties of the transient problem, including stiffness and sparsity, and making detailed use of other features of the system such as certain semilinearity properties (defined below). The proposed methods were applied to a small power system with very encouraging results. With step sizes of the order of 10^{-1} sec. the integration was carried out with good accuracy.

MULTISTEP INTEGRATION METHODS

For a full discussion of these methods the reader is referred to [15]. In what follows we review briefly some facts pertaining to the solution of the initial value problem

$$\dot{\underline{x}} = \underline{F}(\underline{x},t) \quad t \in [t^0,t^f], \ \underline{x}(t^0) = \underline{x}^0 \tag{I.V.P.}$$

where $\underline{x}(t) \in R^{\mu}$ and $\underline{F}: R^{\mu} \times R \to R^{\mu}$ is continuously differentiable. A numerical solution discretizes the differential equation and computes approximations to the exact solution $\underline{x}(\cdot)$ at a sequence of points $t^1, t^2, \ldots, t^N = t^f$. Let us consider a grid with equidistant points t^{ν}, $\nu=0, 1, 2, \ldots, N$ separated by *stepsize* h with $t^{\nu} = t^0 + \nu h$ and $t^N \equiv t^f$. Let $\underline{x}^{\nu} \equiv \underline{x}(t^{\nu})$ be the exact solution of (I.V.P.) at t^{ν}, and denote by $\underline{u}^{\nu}$ the approximation at t^{ν} generated by the computational scheme using infinite precision arithmetic. Numerical integration schemes may be classified according to the amount of information they use: *one-value* or *one-step methods* require the value of the dependent variable at only one grid point to compute the value at the next, whereas *multivalue* or *multistep methods* use the values of the dependent variable and its derivative at several grid points to compute the value at the next. We concern ourselves here exclusively with multistep methods, which constitute an extensive and important class of numerical integration schemes.

After approximations at a number of points, say $t^{\ell}, t^{\ell-1}, \ldots,$ and $t^{\ell+1-k}$, we have values $\underline{u}^{\ell-\nu}$ and also $h\dot{\underline{u}}^{\ell-\nu}$, $\nu=0,1,2,\ldots,k-1$, where $\dot{\underline{u}}^{\ell-\nu} = \underline{F}(\underline{u}^{\ell-\nu}, t^{\ell-\nu})$. A *k-step method* uses this information,

Paper F 76 132-1, recommended and approved by the IEEE Power System Engineering Committee of the IEEE Power Engineering Society for presentation at the IEEE PES Winter Meeting and Tesla Symposium, New York, N.Y., January 25-30, 1976. Manuscript submitted September 3, 1975; made available for printing November 5, 1975.

or a subset of it, to compute $\underline{u}^{\ell+1}$ and has the general form

$$\underline{u}^{\ell+1} = \sum_{\nu=1}^{k} a_\nu \underline{u}^{\ell+1-\nu} + \sum_{\nu=0}^{k} b_\nu h\dot{\underline{u}}^{\ell+1-\nu} \qquad \text{(M.S.)}$$
$$= \sum_{\nu=1}^{k} a_\nu \underline{u}^{\ell+1-\nu} + \sum_{\nu=0}^{k} b_\nu h\underline{F}(\underline{u}^{\ell+1-\nu}, t^{\ell+1-\nu}) .$$

A k-step method is *explicit* if $b_0 = 0$ and *implicit* otherwise. In an explicit scheme $\underline{u}^{\ell+1}$ is easily computed as the linear combination of previously computed values of $\underline{u}^{\ell-\nu}$ and $h\dot{\underline{u}}^{\ell-\nu}$, $\nu=0,1,2,\ldots,k-1$. On the other hand, in an implicit scheme, $\underline{u}^{\ell+1}$ is obtained by solving a set of algebraic equations which are nonlinear if $\underline{F}(\cdot,t)$ is nonlinear. All k-step methods suffer from the shortcoming that they are *not self starting* since initially only t^0 and $\underline{u}^0 = \underline{x}^0$ are known. To compute the starting values $\underline{u}^1, \underline{u}^2, \ldots, \underline{u}^{k-1}$, it is common to use a one-step method such as Runge-Kutta. While the computational efficiency, accuracy, and speed of multistep methods are much superior to those of single step schemes, many practitioners have shied away from multistep methods because of this starting problem.

The *accuracy* of a numerical method refers to the local truncation error resulting from the discretization of the d.e.s. The *local truncation error* (l.t.e.) $\underline{\tau}$ is defined to be the amount by which the value of $\underline{x}^{\ell+1}$ computed with the numerical integration formula using the exact solution data $\underline{x}^{\ell}, \underline{x}^{\ell-1}, \ldots, \underline{x}^{\ell+1-k}$, differs from the exact solution of the d.e. ([15], p. 137). The *order* of a multistep method is the integer m such that the l.t.e. $\underline{\tau}$ is of order $O(h^{m+1})$ ([15], p 117).

An important consideration in a numerical process is how the error at each step affects the results in subsequent steps, i.e. its error propagation ability. This phenomenon is referred to as numerical stability ([15]). The error incurred at each step is the sum of the truncation error and the *roundoff error*. The latter is the error introduced by "rounding" in the arithmetic operations, since in actually performing the calculations, instead of the quantities $\{\underline{u}^\nu\}$ we obtain some quantities $\{\hat{\underline{u}}^\nu\}$. A *numerically stable* algorithm has the desirable property that the accumulated error after ℓ steps $\hat{\underline{u}}^{\ell} - \underline{x}^{\ell}$ does not blow up for large ℓ.

The stability of an integration scheme is discussed with respect to the linear scalar d.e. $\dot{x} = \lambda x$. The *characteristic equation* of the difference relation (M.S.) when applied to this simple system is the k^{th} degree polynomial in ξ

$$\rho(\xi) + h\lambda\,\sigma(\xi) \triangleq \sum_{\nu=0}^{k} a_\nu \xi^{k-\nu} + h\lambda \sum_{\nu=0}^{k} b_\nu \xi^{k-\nu} \qquad \text{(C.E.)}$$

where $a_0 = -1$. The k solutions of (C.E.) are the characteristic roots ξ_ν, $\nu=1,2,\ldots,k$, and the numerical solution in terms of modes is in the form

$$u^{\ell} = \sum_{\nu=1}^{k} c_\nu \xi_\nu^{\ell} .$$

The solution of $\dot{x}=\lambda x$ is $x(t^{\ell}) = c\,e^{\lambda t^{\ell}} = c[e^{h\lambda}]^{\ell}$. One of the characteristic roots, say ξ_1, approximates the exact solution $e^{h\lambda}$ with a truncation error consistent with the order m of the method, i.e. $\xi_1 = \xi_1(h\lambda) = e^{h\lambda} + O(h^{m+1})$. Such a root is called the *principal root*. The other roots ξ_ν giving rise to the *parasitic components* $c_\nu \xi_\nu^{\ell}$ are the *extraneous roots* and are present because the differential equation was replaced by a k^{th} order difference equation. Since the parasitic components contribute errors to the desired solution $c_1 \xi_1^{\ell}$, it is desirable that they not grow with ℓ. A method is called *absolutely stable* for those values of λh for which $|\xi_\nu(\lambda h)| \le 1$, $\nu=1,2,\ldots,k$. It is *relatively stable* for those values of λh where the extraneous roots $\xi_\nu(\lambda h)$, $\nu=2,3,\ldots,k$, satisfy $|\xi_\nu(\lambda h)| \le |\xi_1(\lambda h)|$.

The set of complex numbers λh for which $|\xi_i(\lambda h)| = 1$, $i=1,2,\ldots,k$ forms the stability boundary and its interior is the *region of absolute stability*, i.e. the set of λh for which a k-step method is absolutely stable. Such a region determines, given λ, how big a stepsize h can be taken for the method to be absolutely stable.

For a system of *nonlinear d.e.s.*, the eigenvalues of the Jacobian $\frac{\partial \underline{F}}{\partial \underline{x}}$ are used as the eigenvalues λ_i of the system, since locally, i.e. for sufficiently small h, the behavior of the nonlinear system may be approximated by that of a linear system. A system of d.e.'s is *stiff* if its eigenvalues are widely separated. Such a system gives rise to both fast and slow components corresponding to large and small λ_i, respectively. The fast components control the stability of the method even though these components may have decayed to negligible levels so that the truncation error is governed by the slow components. A comprehensive survey of methods of solution for stiff systems is given in [16]. A class of methods, which exhibit a type of stability termed stiff stability, is very useful for integrating stiff differential systems. Referring to Fig. 1 a method is *stiffly stable* if it is absolutely stable for $h\lambda$ in a region R_I in the complex plane and is accurate for $h\lambda$ in a region R_{II}. Given a particular stiffly stable method, and a set of system eigenvalues λ_i, an h can be chosen such that all the $h\lambda_i$ fall into R_I and/or R_{II}. It is very likely that h can be chosen on the basis of accuracy considerations alone since absolute stability is assured for those eigenvalues very far in the left half plane which are the usual cause of numerical instability. A further discussion of the rationale for the definition and desirability of stiff stability may be found in [12], [15].

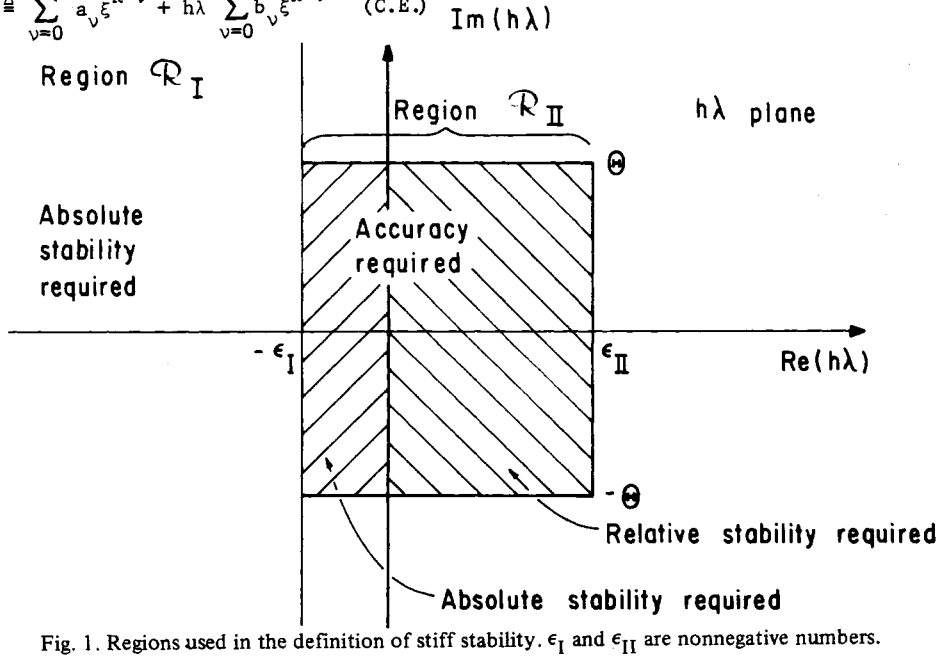


Fig. 1. Regions used in the definition of stiff stability. ϵ_I and ϵ_{II} are nonnegative numbers.

It can be verified that only certain implicit and no explicit multistep schemes can be stiffly stable. Consequently for stiff systems explicit schemes should probably not be used since they could impose very strict requirements on the stepsize h.

STRUCTURE OF THE SYSTEM EQUATIONS

The mathematical model of the n-machine power system is derived in the Appendix. It is in the form

$$\frac{d\underline{i}}{dt} = \underline{A}\,\underline{i} + \underline{b} + \underline{C}(\underline{\delta})\underline{i} \qquad \text{(A15a)}$$

$$\frac{d^2\underline{\delta}}{dt^2} = -\underline{\gamma}\frac{d\underline{\delta}}{dt} + \underline{f}(\underline{i}) \qquad \text{(A15b)}$$

where $\underline{i}$ is a 6n-vector of machine currents, $\underline{\delta}$ is the n-vector of rotor angles. All the matrices are constant except for $\underline{C}(\underline{\delta})$ and $\underline{f}(\underline{i})$. The matrices are given in (A1), (A5), (A6), (A9), (A10), and (A13). Our purpose here is to show how the structural properties of the mathematical model can be exploited to develop a class of efficient integration schemes; eqs. (A15a) and (A15b) clearly show that underlying structure.

The equations have been derived under the assumptions that the mechanical input torques and field voltages are constants. This simplifies the system description and has the advantage that all the matrices in (A15) may be specified explicitly. However these are not crucial assumptions; additional control loops can be included.

It is desirable to retain the differential system in the split form of Eq. (A15) since this mathematical representative is seen to possess some very useful structural properties. The time constants of the electrical system represented by (A15a) range from as short as 0.001 sec. for the amortisseur circuits to as long as 1 sec. for the field circuit. Thus (A15a) is a *stiff* system of differential equations with a conditioning number (the ratio of the longest to the shortest time constant of the system) of about 1000. On the other hand, in the mechanical system represented by (A15b), each second order differential equation describes a lightly damped oscillator with typical periods of several seconds. Then, using the representation in (A15) splits the differential system into a nonstiff part with long time constants [Eq. (A15b)] coupled to a stiff part [Eq. (A15a)] whose longest time constant is usually shorter than that of the first part. Consequently, the split representation results in a smaller dimensional stiff system which may have a smaller conditioning number than the original system.

The differential equations in the split form (A15) are *semilinear*, i.e. the stiff and the nonstiff parts are linear in their respective states. Moreover, the structure of the two subsystems is such that the output of one is the input to the other and vice-versa. A property of the stiff subsystem (A15a) which will be very useful in the computational procedure is that its Jacobian is sparse: basically the Jacobian is the sum of a block diagonal matrix $\underline{A} = \text{diag}\{\underline{A}_\nu : \nu = 1, 2, \ldots, n\}$ and some off-diagonal terms produced by the "coupling terms" $\underline{C}(\underline{\delta})\underline{i}$.

Thus it seems possible to use a longer stepsize and thereby reduce computation time, as compared to current methods, by using the representation (A15) of a multimachine power system. Moreover, both the sparsity and the semilinearity properties can be exploited in the integration process.

DERIVATION OF PROPOSED INTEGRATION ALGORITHMS

In this section we propose a class of methods to integrate (A15) numerically on some interval $[t^0, t^f]$. We denote by $\{\underline{j}^\nu\}$, $\{\underline{s}^\nu\}$, and $\{\underline{u}^\nu\}$ the sequences constructed by these methods using infinite precision arithmetic at the set of points $\{t^\nu : \nu = 0, 1, \ldots, N\}$; these approximate $\{\underline{i}^\nu\}$, $\{\underline{\delta}^\nu\}$, and $\{\underline{\omega}^\nu\}$, respectively, where $\underline{i}^\nu \equiv \underline{i}(t^\nu)$, $\underline{\delta}^\nu \equiv \underline{\delta}(t^\nu)$, $\underline{\omega}^\nu \equiv \underline{\omega}(t^\nu) = \dot{\underline{\delta}}(t^\nu)$, are the exact solutions of (A15) at $t = t^\nu$. This class of methods is derived using a uniform stepsize h.

We start with the mechanical equation (A15b) which we could regard as a second-order linear differential system if $\underline{i}(t)$ were known for all values of t of interest. Suppose that at some instant t^a, $\underline{\delta}(t^a)$, $\underline{\omega}(t^a)$ are known; then for any $t^b \geq t^a$, it can be shown that $\underline{\delta}(t^b)$, $\underline{\omega}(t^b)$ are given by

$$\underline{\delta}(t^b) = \underline{\delta}(t^a) + \underline{\gamma}^{-1}(\underline{U}_n - \exp[-(t^b - t^a)\underline{\gamma}])\underline{\omega}(t^a) + \underline{\gamma}^{-1}\int_{t^a}^{t^b} (\underline{U}_n - \exp[-(t^b - \tau)\underline{\gamma}])\underline{f}(\underline{i}(\tau))\,d\tau \qquad (2)$$

and

$$\underline{\omega}(t^b) = \exp[-(t^b - t^a)\underline{\gamma}]\,\underline{\omega}(t^a) + \int_{t^a}^{t^b} \exp[-(t^b - \tau)\underline{\gamma}]\underline{f}(\underline{i}(\tau))\,d\tau, \qquad (3)$$

where $\underline{U}_n$ is the n-dimensional identity matrix. Then, with $t^a = t^\ell$ and $t^b = t^{\ell+1} = t^\ell + h$, (2) becomes

$$\underline{\delta}^{\ell+1} = \underline{\delta}^\ell + \underline{\gamma}^{-1}[\underline{U}_n - \exp(-\underline{\gamma}h)]\underline{\omega}^\ell + \underline{\gamma}^{-1}\int_{t^\ell}^{t^{\ell+1}} (\underline{U}_n - \exp[-(t^{\ell+1} - \tau)\underline{\gamma}])\underline{f}(\underline{i}(\tau))\,d\tau \qquad (4)$$

After approximations at a number of points, say $t^{\ell-k+1}, t^{\ell-k+2}, \ldots,$ and t^ℓ have been calculated, we have values of $\underline{j}^{\ell-k+1}, \underline{j}^{\ell-k+2}, \ldots, \underline{j}^\ell$, $\underline{s}^{\ell-k+1}, \ldots, \underline{s}^\ell$, $\underline{u}^{\ell-k+1}, \ldots,$ and $\underline{u}^\ell$. The computation of $\underline{s}^{\ell+1}$, the approximation of $\underline{\delta}^{\ell+1}$, is based on the relation (4). We approximate $\underline{\delta}^\ell$ and $\underline{\omega}^\ell$ by $\underline{s}^\ell$ and $\underline{u}^\ell$, and the integral by numerical quadrature. The unique k^{th} degree polynomial interpolating the integrand $(\underline{U}_n - \exp[-(t^{\ell+1} - t)\underline{\gamma}])\,\underline{f}(\underline{i}(t))$ at the (k+1) points $t^{\ell+1}, t^\ell, \ldots, t^{\ell+1-k}$ is used to construct $\sum_{\nu=0}^{k} \sigma_{k,\nu}(\underline{U}_n - \exp[-(t^{\ell+1} - t^{\ell+1-\nu})\underline{\gamma}])\,\underline{f}(\underline{i}^{\ell+1-\nu})$, the interpolatory quadrature formula for the integral in (4). The coefficients $\sigma_{k,\nu}$ are constant and are readily evaluated for equidistant interpolation points [12]; a table of coefficients is provided in Table I. Thus we compute $\underline{s}^{\ell+1}$ by

$$\underline{s}^{\ell+1} = \underline{s}^\ell + \underline{\gamma}^{-1}[\underline{U}_n - \exp(-\underline{\gamma}h)]\underline{u}^\ell + \underline{\gamma}^{-1}\sum_{\nu=0}^{k}\sigma_{k,\nu}[\underline{U}_n - \exp(-\underline{\gamma}\nu h)]\underline{f}(\underline{j}^{\ell+1-\nu}) \qquad (5)$$

Note that the computation of the value of $\underline{\delta}$ at time $t^{\ell+1}$ by (4) requires that value of $\underline{i}$ at $t^{\ell+1}$, i.e. (4) computes $\underline{\delta}(\cdot)$ using an 'implicit-type' scheme. We now make the important observation that the computation of $\underline{s}^{\ell+1}$ by (5) does *not* require knowledge of $\underline{j}^{\ell+1}$ since the coefficient of $\underline{f}(\underline{j}^{\ell+1})$ is $\sigma_{k,0}[\underline{U}_n - \exp(\underline{0})] = \underline{0}$. Consequently, the lower limit of the summation in (5) becomes 1 so that *using (5) we compute* $\underline{s}^{\ell+1}$ *explicitly.*

We next consider the electrical system (A15a). Note that if $\underline{\delta}(t)$ were known for all values of t, then $\underline{i}(t)$ would be the solution of a linear time varying system. Our method to solve (A15a) is based on numerical differentiation. This differs from the use of convolution in (A15b). In principle the same approach could be used here but we wish to avoid the use of a high-dimensional state transition matrix.

After approximations at points $t^{\ell-k+1}, t^{\ell-k+2}, \ldots, t^\ell$ have been calculated, we have values of $\underline{j}^{\ell-k+1}, \underline{j}^{\ell-k+2}, \ldots, \underline{j}^\ell$, $\underline{s}^{\ell-k+1}, \ldots, \underline{s}^\ell$, $\underline{u}^{\ell-k+1}, \ldots,$ and $\underline{u}^\ell$. Then $\underline{s}^{\ell+1}$ may be computed using (5). We use the derivative of the interpolation polynomial through $\{(t^{\ell+1-\nu}, \underline{i}^{\ell+1-\nu}) : \nu = 0, 1, 2, \ldots, k\}$ to approximate $\frac{d\underline{i}}{dt}$ at $t = t^{\ell+1}$.

In (A15a) we replace this derivative by $\frac{1}{h\mu_{k,0}}[\underline{i}^{\ell+1} - \sum_{\nu=1}^{k} \mu_{k,\nu}\underline{i}^{\ell+1-\nu}]$. The backward differentiation coefficients are provided in table II. Upon replacing the exact values of the state variables by their approximations we obtain the implicit computational scheme for $\underline{j}^{\ell+1}$

$$\underline{j}^{\ell+1} = h\mu_{k,0}\{[\underline{A}+\underline{C}(\underline{s}^{\ell+1})]\underline{j}^{\ell+1}+\underline{b}\} + \sum_{\nu=1}^{k}\mu_{k,\nu}\underline{j}^{\ell+1-\nu} \quad (6)$$

Upon rearranging, we have

$$\{\underline{U}-h\mu_{k,0}[\underline{A}+\underline{C}(\underline{s}^{\ell+1})]\}\ \underline{j}^{\ell+1} = h\mu_{k,0}\underline{b} + \sum_{\nu=1}^{k}\mu_{k,\nu}\underline{j}^{\ell+1-\nu}. \quad (7)$$

where $\underline{U}$ is the 6n dimensional identy matrix. Note that the Jacobian of (A15a) is sparse as it is the sum of a block diagonal matrix $\underline{A}$ and a matrix $\underline{C}(\cdot)$ which has a few nonzero entries in only one third of its columns. Clearly, the coefficient matrix $\{\underline{U} - h\mu_{k,0}[\underline{A} + \underline{C}(\underline{s}^{\ell+1})]\}$ is also sparse. Consequently, we may solve rapidly for $\underline{j}^{\ell+1}$ in (7) using optimally ordered triangular factorization and packed storage schemes [10].

Finally, we derive a scheme which calculates $\underline{u}^{\ell+1}$. We use (3) to compute $\underline{\omega}^{\ell+1}$ once $\underline{\omega}^{\ell}$ is known:

$$\underline{\omega}^{\ell+1} = \exp(-\underline{\gamma}h)\underline{\omega}^{\ell} + \int_{t^{\ell}}^{t^{\ell+1}} \exp[-(t^{\ell+1}-\tau)\underline{\gamma}]\underline{f}(\underline{i}(\tau))d\tau \quad (8)$$

We approximate $\underline{\omega}^{\ell}$ by $\underline{u}^{\ell}$, $\underline{i}^{\ell+1-\nu}$ by $\underline{j}^{\ell+1-\nu}$, ν=0,1,2,...,k, and evaluate the integral in (8) using numerical quadrature. Proceeding as we have from (4) to (5) we can compute $\underline{u}^{\ell+1}$ by

$$\underline{u}^{\ell+1} = \exp(-\underline{\gamma}h)\underline{u}^{\ell} + \sum_{\nu=0}^{k}\sigma_{k,\nu}\exp(-\underline{\gamma}\nu h)\underline{f}(\underline{j}^{\ell+1-\nu}) \quad (9)$$

We can summarize the above discussion in the following algorithm to integrate the differential system (A15).

ALGORITHM: k-step integration routine for solving (A15) for $t \in [t^0, t^f]$.

Step 0. The quantities $\underline{j}^0,\underline{j}^1,\ldots,\underline{j}^{k-1},\underline{s}^0,\underline{s}^1,\ldots,\underline{s}^{k-1}$, $\underline{u}^0,\underline{u}^1,\ldots$, and $\underline{u}^{k-1}$ at the time points $t^0,t^1,\ldots,t^{k-1}$ are given. Set $\ell = k-1$, and $t^{\ell+1} = t^{\ell} + h$.

Step 1. Compute the coefficients $\sigma_{k,\nu},\mu_{k,\nu},\nu=0,1,2,\ldots,k$, [see Tables I and II]. Compute recursively $[\underline{\Lambda}(h)]^{\nu-1}\underline{\Lambda}(h)$ for $\nu=1,2,3,\ldots,k$ where

$$\underline{\Lambda}(h) \triangleq \exp(-\underline{\gamma}h) = \mathrm{diag}\{e^{-\gamma_{\mu}h}:\ \mu=1,2,\ldots,n\} \quad (10)$$

and $[\underline{\Lambda}(h)]^0 \equiv \underline{U}_n$.

Step 2. Compute $\underline{s}^{\ell+1}$

$$\underline{s}^{\ell+1} = \underline{s}^{\ell} + \underline{\gamma}^{-1}[\underline{U}_n - \underline{\Lambda}(h)]\underline{u}^{\ell} + \underline{\gamma}^{-1}\sum_{\nu=1}^{k}\sigma_{k,\nu}(\underline{U}_n - [\underline{\Lambda}(h)]^{\nu})\underline{f}(\underline{j}^{\ell+1-\nu}) \quad (11)$$

Step 3. Solve for $\underline{j}^{\ell+1}$

$$\{\underline{U}-h\mu_{k,0}[\underline{A}+\underline{C}(\underline{s}^{\ell+1})]\}\underline{j}^{\ell+1} = h\mu_{k,0}\underline{b} + \sum_{\nu=1}^{k}\mu_{k,\nu}\underline{j}^{\ell+1-\nu} \quad (12)$$

Step 4. Compute $\underline{u}^{\ell+1}$

$$\underline{u}^{\ell+1} = \underline{\Lambda}(h)\underline{u}^{\ell} + \sum_{\nu=0}^{k}\sigma_{k,\nu}[\underline{\Lambda}(h)]^{\nu}\underline{f}(\underline{j}^{\ell+1-\nu}). \quad (13)$$

Step 5. If $t^{\ell+1} = t^f$, stop; else set $\ell = \ell+1$, $t^{\ell+1} = t^{\ell+1} + h$ and go to step 2.

COMMENTS ON THE PROPOSED CLASS OF ALGORITHMS

The proposed method is a k-step integration scheme as it uses the values of $\underline{s}$, $\underline{j}$ and $\underline{u}$ at $t^{\ell}, t^{\ell-1},\ldots,t^{\ell-(k-1)}$, to compute $\underline{s}^{\ell+1}$, $\underline{j}^{\ell+1}$, and $\underline{u}^{\ell+1}$ (at $t^{\ell+1}$). To apply this method, the *initial values* $\underline{s}^0,\underline{j}^0,\underline{u}^0$ and the *starting values* $\underline{s}^{\ell},\underline{j}^{\ell},\underline{u}^{\ell}$, $\ell = 1, 2, \ldots, k-1$ are required. The latter may be obtained by applying successively this scheme with "k" = 1, 2, ..., k-1. Thus the proposed integration algorithm is *self starting*, thereby avoiding the "starting problem" usually associated with multistep schemes.

It can be shown [12] that the local truncation errors of the integration formulae (11), (12) and (13) are

$$\underline{\tau}_{\underline{i}}^{\ell+1} = \underline{i}^{\ell+1}\text{ given by (12)} - \underline{i}^{\ell+1} = O(h^{k+1})$$

$$\underline{\tau}_{\underline{\delta}}^{\ell+1} = \underline{\delta}^{\ell+1}\text{ given by (11)} - \underline{\delta}^{\ell+1} = O(h^{k+2})$$

$$\underline{\tau}_{\underline{\omega}}^{\ell+1} = \underline{\omega}^{\ell+1}\text{ given by (13)} - \omega^{\ell+1} = O(h^{k+2})$$

so that the proposed integration formulas are of order k for the electrical part and of order k+1 for the mechanical part.

The quadrature coefficients $\sigma_{k,\nu}$ were calculated for $1 \le k \le 5$ and are presented in Table I. They are, incidentally, the same as the coefficients of the (implicit) Adams-Moulton integration scheme [15].

TABLE I

Interpolatory quadrature coefficients $\sigma_{k,\nu}$

$\sigma_{k,\nu}$ \ ν	0	1	2	3	4	5
$2h^{-1}\sigma_{1,\nu}$	1	1				
$12h^{-1}\sigma_{2,\nu}$	5	8	-1			
$24h^{-1}\sigma_{3,\nu}$	9	19	-5	1		
$720h^{-1}\sigma_{4,\nu}$	251	646	-264	106	-19	
$1440h^{-1}\sigma_{5,\nu}$	475	1427	-798	482	-173	27

Note that since $\underline{\gamma}$ and consequently $\underline{\Lambda}$ are diagonal matrices we can immediately replace (11) and (13) by n scalar equations and obtain some simplification.

Equation (12) is used to integrate the stiff differential equation (A15a), and has the desirable property that, for $k \leq 6$, it is stiffly stable. Fig. 2 shows the regions of absolute stability of (12) for $1 \leq k \leq 5$. Regions R_I can be found, (see Fig. 1) by picking $\epsilon_I = 0$ for $k = 1, 2$, $\epsilon_I = 0.1$ for $k = 3$, $\epsilon_I = 0.7$ for $k = 4$, and $\epsilon_I = 2.4$ for $k = 5$. For $k = 3, 4, 5$ the region R_{II} is obtained by requiring that its left half plane portion lie in the region of absolute stability.

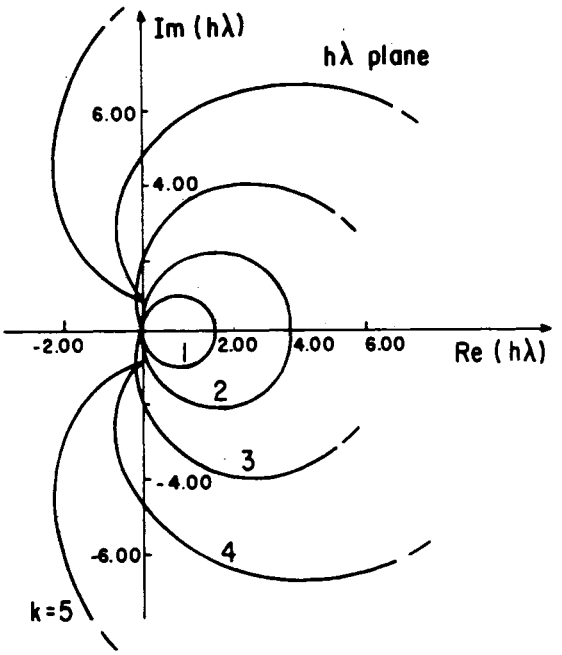


Fig. 2. The regions of absolute stability are outside the closed contours.

Table II gives the coefficients $\mu_{k,\nu}$ for $1 \leq k \leq 5$. Since $\mu_{k,0} \neq 0$, the backward differentiation schemes are implicit multistep methods (see (6)). But, because the implicit equations are linear, they may be solved in only one iteration. Moreover, the linear system has the sparse coefficient matrix $\{\underline{U} - h\mu_{k,0}[\underline{A} + \underline{C}]\}$, (see (11)), and can be solved by optimally ordered triangular factorization [10]. Note that since the structure of this matrix remains unaltered for a fixed network configuration, only one row and column reordering is necessary for the integration period between any two consecutive discontinuities. Furthermore $\underline{\delta}$ changes much slower than $\underline{i}$, so that the coefficient matrix may be kept constant for several iterations thereby further reducing the computational effort.

TABLE II

Backward differentiation coefficients $\mu_{k,\nu}$

$\mu_{k,\nu}$ \ ν	0	1	2	3	4	5
$\mu_{1,\nu}$	1	1				
3 $\mu_{2,\nu}$	2	4	-1			
11 $\mu_{3,\nu}$	6	18	-9	2		
25 $\mu_{4,\nu}$	12	48	-36	16	-3	
137 $\mu_{5,\nu}$	60	300	-300	200	-75	12

We note that for $k = 1$, then, we use the *trapezoidal rule* for numerical quadrature and the *backward Euler* method for backward differentiation. A possible modification is to replace the latter by the trapezoidal rule which is a stiffly stable method [15]. Analytic bounds on the accumulated error for the two $k = 1$ schemes have been obtained, with and without roundoff [12].

EXAMPLE

We tested the proposed class of algorithms on the six bus four generator power system shown in Fig. 3. This system was chosen to be almost identical to that considered by Talukdar [7], [17]. For a description and the necessary data the reader is referred to the above references. We simulated the response of the system to a step increase of the load admittance at bus 6 by a factor of 1.75. The damping coefficients were changed from 0.0 in [7] to the more realistic value of 2.0.

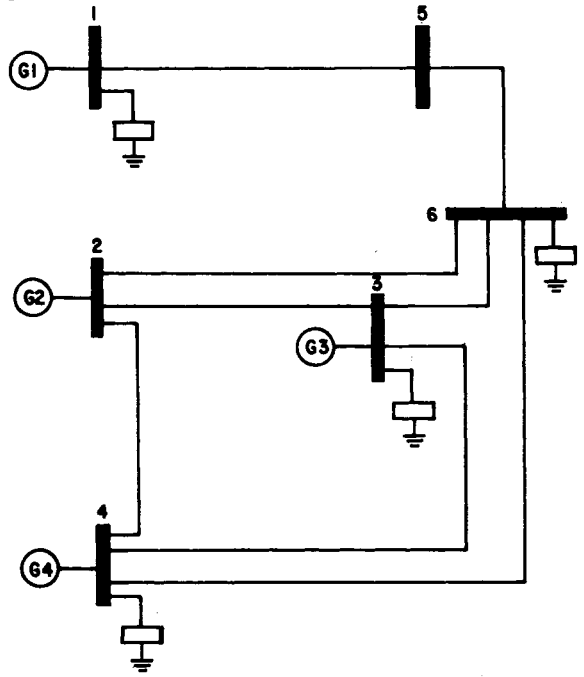
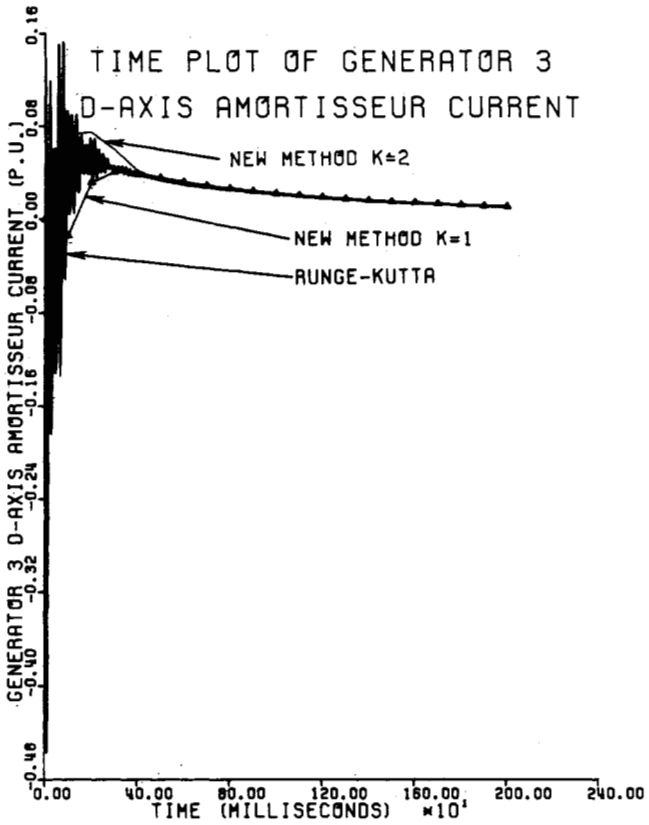
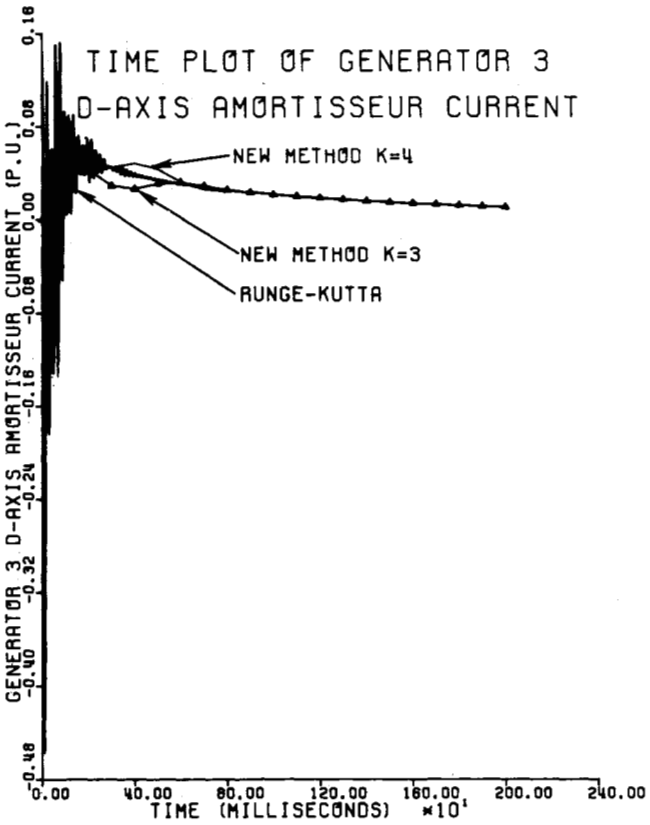


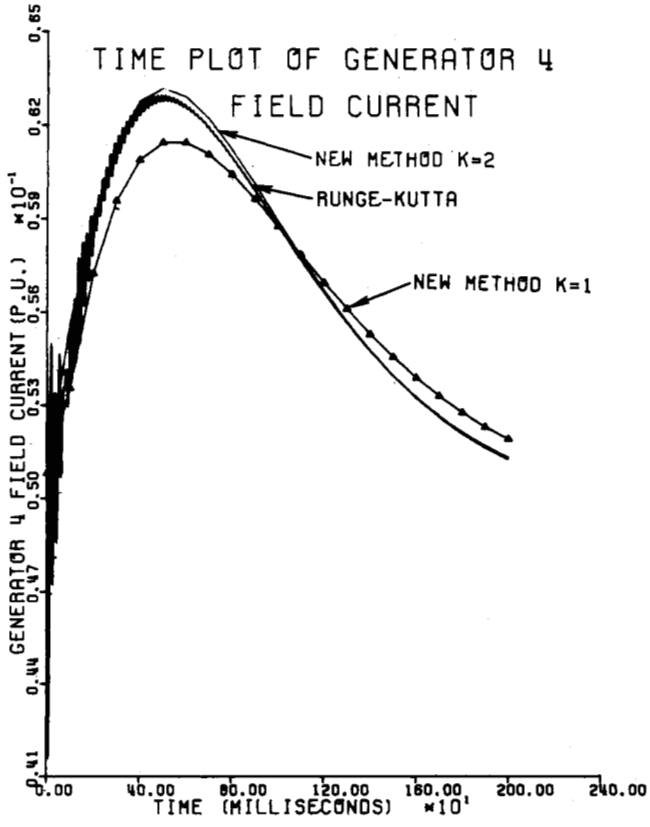
Fig. 3. Single line diagram of system studied.

We integrated from 0 to 2 sec. using the algorithm with $k = 1, 2, 3, 4$ and also the fourth order Runge-Kutta with a 10^{-4} sec. stepsize. To determine the accuracy of the proposed algorithms we compared the values of the integrated variables at $t = 2$ sec. to those obtained using the Runge-Kutta scheme; in what follows we refer to the difference between the two as the error and to the maximum of the absolute value of its components as its (ℓ_∞) norm. For $k = 2, 3, 4$ the norm of the error was of the order 10^{-4}-10^{-3} p.u.; for $k = 1$, the error was of the order 10^{-2}. The sample plots in Fig. 4 were obtained with the new integration schemes using a 10^{-1} sec. stepsize; the Runge-Kutta integration results (with a stepsize of 10^{-4} sec.) are also presented for comparison. After about 0.5 secs., the curves, for $k > 1$, are in very close agreement. Of course, at the very beginning, no integration scheme with a stepsize of 0.1 sec. can show the high frequency oscillations. But they are being considered correctly and hence the accuracy of the remainder of the plot.

Talukdar applied his modified multistep predict or corrector schemes [7] to this system for the identical distrubance and reported good accuracy. However because of numerical stability the largest stepsize possible was of the order 10^{-3}. Indeed, this stepsize barrier is also encountered with the fourth order Runge-Kutta method: we obtained numerical instability with stepsizes larger than 0.003 sec. With the new methods we could use stepsizes even 100 times as large with no numerical stability problems and without sacrificing accuracy. Finally it is worth noting that Runge-Kutta schemes require more computations at each time step than the new methods.







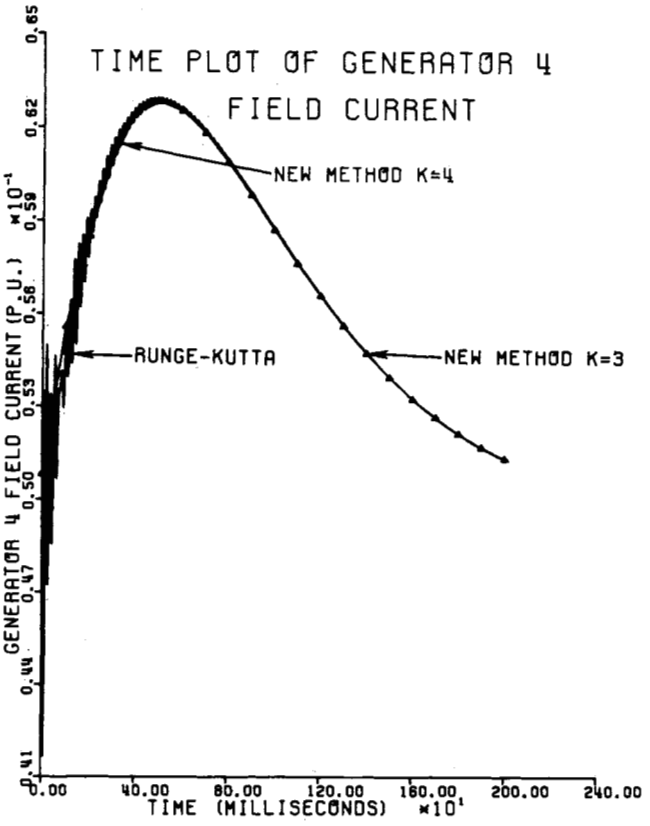


Fig. 4. Plots of i_{kd3} and i_{f4} versus time. For the proposed schemes the stepsize is 10^{-1} sec. while for the Runge-Kutta method the stepsize is 10^{-4} sec.

CONCLUSION

In this paper we reported on the development of a class of efficient numerical integration schemes for computing power system dynamic response. We derived these schemes by making detailed use of the structural properties of the mathematical model of the multimachine power network. The results of numerical experiments indicate that the proposed methods integrate the system equations using much larger stepsizes with good accuracy after the initial high frequency transient and consequently offer significant savings in the computational effort required when compared to present methods. An important feature of the new schemes is that although they are implicit methods no iteration at each time step is necessary. This is an improvement over previous work which found that a limiting factor to using larger stepsizes when multistep integration methods were applied to the study of power system stability was the increasing number of corrector iterations [6], [7].

Some extension of the work reported here has already been made. A modification of the BPA Transients Program to incorporate the simulation of machine transients based on the methods of this paper is now undergoing final testing.

Further work along the lines reported in this paper remains to be done. For example, we did not take into consideration speed governor and voltage regulator effects since the purpose here was to show that the structural properties of the differential equations describing the multimachine power system dynamic behavior could be exploited to develop efficient computational schemes. We foresee no difficulties in extending these techniques to include the auxiliary control systems, or, for that matter, boiler dynamics. Furthermore, the methods outlined here allow the representation of as many rotor circuits as available data permit, to account for *dynamic saliency* [3]. We can also introduce the effect of magnetic saturation when the latter are represented as functions of the rotor flux linkages or the rotor currents. With these additions the proposed algorithms are expected to furnish a valuable tool for the investigation of the long duration transients which are of current interest.

ACKNOWLEDGMENT

Research sponsored in part by the National Science Foundation Grants GK-10656X3/ENG74-06651-A01.

REFERENCES

(1) G. Shackshaft, "General-purpose turbo-alternator model," Proc. IEE, Vol. 110, No. 4, pp. 703-713, April 1963.

(2) W. D. Humpage and B. Stott, "Predictor-corrector methods of numerical integration in digital computer analysis of power system transient stability," Proc. IEE, Vol. 112, pp. 1557-1565, Aug. 1965.

(3) J. M. Undrill, "Structure in the computation of power-system nonlinear dynamical response," IEEE Trans. Power Apparatus and Systems, Vol. PAS-88, pp. 1-6, January 1967.

(4) K. Prabhashankar and W. Janischewskyj, "Digital simulation of multimachine power systems for stability studies," IEEE Trans. Power Apparatus and Systems, Vol. PAS-87, pp. 73-80, January 1968.

(5) D. W. Olive, "Digital simulation of synchronous machine transients," IEEE Trans. Power Apparatus and Systems, Vol. PAS-87, pp. 1669-1675, Aug. 1968.

(6) K. N. Stanton and S. N. Talukdar, "New integration algorithms for transient stability studies," IEEE Trans. Power Apparatus and Systems, Vol. PAS-89, pp. 985-991, May/June 1970.

(7) S. N. Talukdar, "Iterative multistep methods for transient stability studies," IEEE Trans. Power Apparatus and Systems, Vol. PAS-90, pp. 96-102, Jan./Feb. 1971.

(8) R. Oswald and O. J. M. Smith, "Integration of nonlinear differential systems with wide eigenvalue range," IEEE Trans. Power Apparatus and Systems, Vol. PAS-90, pp. 2586-2589, Nov./Dec. 1971.

(9) H. W. Dommel and N. Sato, "Fast transient stability solutions," IEEE Trans. Power Apparatus and Systems, Vol PAS-91, pp. 1643-1650, Jul./Aug. 1972.

(10) W. F. Tinney and W. S. Meyer, "Solution of large sparse systems by ordered triangular factorization," IEEE Trans. Automat. Control, Vol. AC-18, pp. 333-346, Aug. 1973.

(11) P. M. Hirsch et at., "Analytical techniques for transient stability," final report submitted to BPA by the Federal Systems Division, International Business Machines, Houston, Texas, 1972.

(12) G. Gross, "Transient stability of multimachine power systems," Ph.D. thesis, University of California, Berkeley, May 1974.

(13) W. D. Humpage et al., "Numerical integration algorithms in power-system dynamic analysis," Proc. IEE, vol. 121, pp. 467-473, June 1974.

(14) V. Converti et al., "Long-term stability of interconnected power systems," paper F75 444-0 presented at the IEEE PES Summer Meeting, San Francisco, July 1975.

(15) C. W. Gear, Numerical Initial Value Problems in Ordinary Differential Equations. Englewood Cliffs, New Jersey: Prentice-Hall, 1971.

(16) G. Bjurel, G. Dahlquist, B. Lindberg, S. Linde, and L. Odén, "Survey of stiff ordinary differential equations," Rept. NA 70.11, Dept. of information Processing, The Royal Institute of Technology, Stockholm, Sweden.

(17) S. N. Talukdar, "Multistep and modified integration methods for the simulation of power system dynamics," Ph.D. dissertation, Purdue University, Lafayette, In., Jan. 1970.

APPENDIX: DERIVATION OF SYSTEM EQUATIONS

Machine Model. We use the well known Park's equations to represent the synchronous machines. Let

$$\theta(t) \triangleq \omega_s t + \delta - \frac{\pi}{2} \ , \quad \dot{\theta}(t) = \omega_s + \dot{\delta}$$

be the rotor angular position and velocity, respectively, where δ, is the rotor angle. Under a common set of assumptions, the Park's equations are [1], [5]:

$$\left.\begin{aligned} v_f &= \dot{\lambda}_f + R_f i_f \\ v_d &= \dot{\lambda}_d - R_a i_d - \omega_s \lambda_q \\ 0 &= \dot{\lambda}_{kd} + R_{kd} i_{kd} \end{aligned}\right\} \text{direct axis voltage relations}$$

$$\left.\begin{aligned} 0 &= \dot{\lambda}_g + R_g i_g \\ v_q &= \dot{\lambda}_q - R_a i_q + \omega_s \lambda_d \\ 0 &= \dot{\lambda}_{kq} + R_{kq} i_{kq} \end{aligned}\right\} \text{quadrature axis voltage relations}$$

$$\begin{bmatrix} \lambda_f \\ \lambda_d \\ \lambda_{kd} \end{bmatrix} = \begin{bmatrix} L_f & -L_{af} & L_{fkd} \\ L_{af} & -L_d & L_{akd} \\ L_{fkd} & -L_{akd} & L_{kd} \end{bmatrix} \begin{bmatrix} i_f \\ i_d \\ i_{kd} \end{bmatrix} \triangleq \underline{L}_d \begin{bmatrix} i_f \\ i_d \\ i_{kd} \end{bmatrix}$$

$$\begin{bmatrix} \lambda_g \\ \lambda_q \\ \lambda_{kq} \end{bmatrix} = \begin{bmatrix} L_g & - & L_{gkq} \\ L_{ag} & -L_q & L_{akq} \\ L_{gkq} & -L_{akq} & L \end{bmatrix} \begin{bmatrix} i_g \\ i_q \\ i_{kq} \end{bmatrix} \triangleq \underline{L}_q \begin{bmatrix} i_g \\ i_q \\ i_{kq} \end{bmatrix}$$

It is useful to define the arrays

$$\underline{\lambda} \triangleq [\lambda_f, \lambda_d, \lambda_{kd}, \lambda_g, \lambda_q, \lambda_{kq}]^T,$$

$$\underline{i} \triangleq [i_f, i_d, i_{kd}, i_g, i_q, i_{kq}]^T,$$

$$\underline{v} \triangleq [v_f, v_d, 0, 0, v_q, 0]^T, \tag{A1}$$

$$\underline{R} \triangleq \text{diag}\{R_f, -R_a, R_{kd}, R_g, -R_a, R_{kq}\},$$

$$\underline{L} \triangleq \text{diag}\{\underline{L}_d, \underline{L}_q\}$$

and

$$\underline{Q} \triangleq \left[\begin{array}{c|c} \underline{0}_{3\times 3} & \underline{Q}_{dq} \\ \hline \underline{Q}_{qd} & \underline{0}_{3\times 3} \end{array}\right], \text{ where } \underline{Q}_{dq} = -\underline{Q}_{qd}^T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Then, for machine ν of an n-machine interconnection the Park machine equations become

$$\underline{v}_\nu = \dot{\underline{\lambda}}_\nu + \underline{R}_\nu \underline{i}_\nu + \omega_s \underline{Q}\, \underline{\lambda}_\nu \tag{A2a}$$

$$\nu = 1, 2, \ldots, n$$

$$\underline{\lambda}_\nu = \underline{L}_\nu \underline{i}_\nu \tag{A2b}$$

The mechanical equations of motion describing the behavior of the νth rotor angle is given by

$$J_\nu \frac{d^2\delta_\nu}{dt^2} = T_\nu^m - T_\nu^e - D_\nu \frac{d\delta_\nu}{dt} \tag{A3}$$

T_ν^e, the electrical air gap torque is given by

$$T_\nu^e = \lambda_{d\nu}\, i_{q\nu} - \lambda_{q\nu}\, i_{d\nu}. \tag{A4}$$

Network Model. We follow common practice, assume loads are constant impedances, and represent the network interconnection as a lumped linear passive network. The real and imaginary parts of the machine terminal voltages and currents are then related by

$$\underline{V}^N = \underline{Z}^N \underline{I}^N. \tag{A5}$$

where $\underline{Z}^N$ is the reduced 2nx2n real bus impedance matrix of the network with all nonmachine nodes eliminated. In the usual way, the network variables in (A5) and the machine variables in (A2)–(A4) are related by sets of orthogonal transformations

$$\underline{T}(\delta_\nu) \triangleq \begin{bmatrix} \cos\delta_\nu & \sin\delta_\nu \\ -\sin\delta_\nu & \cos\delta_\nu \end{bmatrix}, \tag{A6}$$

and in terms of machine variables, (A5) becomes

$$\underline{v}^m = \underline{Z}(\underline{\delta})\, \underline{i}^m \tag{A7}$$

where the 2n-vectors $\underline{v}^m$ and $\underline{i}^m$ are given by

$$\underline{v}^m = \text{col}\{\underline{v}_\nu^m: \ \nu=1,n\}, \ \underline{v}_\nu^m = [v_{d\nu}, v_{q\nu}]^T$$

$$\underline{i}^m = \text{col}\{\underline{i}_\nu^m: \ \nu=1,n\}, \ \underline{i}_\nu^m = [i_{d\nu}, i_{q\nu}]^T$$

and

$$\underline{Z}(\underline{\delta}) = \left[\begin{array}{c|c|c} \underline{Z}_{11}(\underline{\delta}) & \cdots & \underline{Z}_{1n}(\underline{\delta}) \\ \hline \vdots & & \vdots \\ \hline \underline{Z}_{n1}(\underline{\delta}) & \cdots & \underline{Z}_{nn}(\underline{\delta}) \end{array}\right] = \left[\begin{array}{c} \underline{Z}_1(\underline{\delta}) \\ \hline \vdots \\ \hline \underline{Z}_n(\underline{\delta}) \end{array}\right] \tag{A8}$$

Here $\underline{Z}_{ij}(\underline{\delta}_i) = \underline{T}(\delta_i)\underline{Z}_{ij}^N \underline{T}^{-1}(\delta_j)$ and $\underline{Z}_{ij}^N$ is the i,jth 2x2 submatrix of $\underline{Z}^N$. The vectors $\underline{i}^m$ and $\underline{v}^m$ are, of course, related to the local machine vectors $\underline{i}_\nu$ and $\underline{v}_\nu$ previously defined. If we form $\underline{i} \triangleq \text{col}\{\underline{i}_\nu: \ \nu=1,2,\ldots n\}$, and $\underline{v} \triangleq \text{col}\{\underline{v}_\nu: \ \nu=1,2,\ldots n\}$ then $\underline{i}^m = \underline{E}\,\underline{i}$, where $\underline{E} \triangleq \{\text{diag}\,\underline{E}_\nu: \nu=1,2,\ldots n\}$ and $\underline{E}_\nu \triangleq \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$.

Then $\underline{E}_\nu$ is an elimination and consolidation operator. Similarly $\underline{v}^m = \underline{E}\,\underline{v}$. With $\underline{u}_1 = [1,0,0,0,0,0]^T$ we have

$$\begin{aligned} \underline{v}_\nu &= v_{f\nu}\, \underline{u}_1 + \underline{E}_\nu^T \underline{v}_\nu^m \\ &= v_{f\nu}\, \underline{u}_1 + \underline{E}_\nu^T \underline{Z}_\nu(\underline{\delta})\, \underline{i}^m \\ &= v_{f\nu}\, \underline{u}_1 + \underline{E}_\nu^T \underline{Z}_\nu(\underline{\delta})\, \underline{E}\, \underline{i} \end{aligned} \tag{A9}$$

Finally, combining (A2a), (A2b) and (A9) we get

$$\begin{aligned} \frac{d\underline{i}_\nu}{dt} = &-(\underline{L}_\nu^{-1}\underline{R}_\nu + \omega_s \hat{\underline{Q}}_\nu)\underline{i}_\nu + \underline{L}_\nu^{-1}\underline{u}_1 v_{f\nu} \\ &+ \underline{L}_\nu^{-1}\underline{E}_\nu^T \underline{Z}_\nu(\underline{\delta})\, \underline{E}\, \underline{i}, \ \nu=1,2,..,n \end{aligned} \tag{A10}$$

where $\hat{\underline{Q}}_\nu \triangleq \underline{L}_\nu^{-1}\, \underline{Q}\, \underline{L}_\nu$.

Next the electrical torque equation (A4) may be expressed in terms of $\underline{i}_\nu$, i.e.

$$T_\nu^e = \underline{i}_\nu^T \underline{Q}\, \underline{L}_\nu\, \underline{i}_\nu \qquad \nu=1,2,\ldots,n \tag{A11}$$

The mechanical equation (A3) becomes

$$\frac{d^2\delta_\nu}{dt^2} = -\frac{D_\nu}{J_\nu}\frac{d\delta_\nu}{dt} + \frac{1}{J_\nu}(T_\nu^m - \underline{i}_\nu^T \underline{Q}\, \underline{L}_\nu\, \underline{i}_\nu)$$

With $\nu=1,2,\ldots,n$ (A12)

$$\underline{A}_\nu = -(\underline{L}_\nu^{-1}\underline{R}_\nu + \omega_s \hat{\underline{Q}}_\nu), \ \underline{A} = \text{diag}\{\underline{A}_\nu: \ \nu=1,2,..,n\}$$

$$\underline{b}_\nu = v_{f\nu}\, \underline{L}_\nu^{-1}\, \underline{u}_1 \qquad , \ \underline{b} = \text{col}\{\underline{b}_\nu: \ \nu=1,2,..,n\}$$

$$\underline{C}_\nu(\underline{\delta}) = \underline{L}_\nu^{-1}\underline{E}_\nu^T \underline{Z}_\nu(\underline{\delta})\, \underline{E} \ , \ \underline{C}(\underline{\delta}) = \text{diag}\{\underline{C}_\nu(\underline{\delta}): \ \nu=1,2,..n\}$$

$$f_\nu(\underline{i}_\nu) = \frac{1}{J_\nu}(T_\nu^m - \underline{i}_\nu^T \underline{Q}\, \underline{L}_\nu \underline{i}_\nu), \ \underline{f}(\underline{i}) = \text{col}\{f_\nu(\underline{i}_\nu): \ \nu=1,2,..n\}$$

$$\gamma_\nu = \frac{D_\nu}{J_\nu} \qquad , \ \underline{\gamma} = \text{diag}\{\gamma_\nu; \ \nu=1,2,\ldots n\} \tag{A13}$$

The differential-algebraic system (A2) - (A6) has been reduced to the differential system

$$\frac{d\underline{i}_\nu(t)}{dt} = \underline{A}_\nu \underline{i}_\nu(t) + \underline{b}_\nu + \underline{C}_\nu(\underline{\delta}(t))\, \underline{i}(t) \tag{A14a}$$

and $\nu=1,2,3,\ldots,n$

$$\frac{d^2\delta_\nu(t)}{dt^2} = -\gamma_\nu \frac{d\delta_\nu(t)}{dt} + f_\nu(\underline{i}_\nu(t)) \tag{A14b}$$

or equivalently

$$\frac{d\underline{i}(t)}{dt} = \underline{A}\, \underline{i}(t) + \underline{b} + \underline{C}(\underline{\delta}(t))\, \underline{i}(t) \tag{A15a}$$

$$\frac{d^2\underline{\delta}(t)}{dt^2} = -\underline{\gamma}\frac{d\underline{\delta}(t)}{dt} + \underline{f}\,(\underline{i}(t)). \tag{A15b}$$

Then (A15a) gives the equations of motion on the electrical side, and (A15b) describes the dynamics of the mechanical side of the n-machine interconnection.

Discussion

P. M. Hirsch (IBM Corporation, Palo Alto, CA): I feel, as do Gross and Bergen in this interesting paper, that multistep implicit integration algorithms are an important improvement over fixed step explicit algorithms, particularly for simulation times exceeding 1-2 seconds past the fault.

Multistep implicit integration methods have been reported previously (11, 14, x). The authors foresee no difficulties in extending their version of the algorithm to include the auxiliary control systems. As discussed in Fuller, Hirsch, and Lambie (x), this extension is the most difficult aspect of application of the algorithms, particularly when there are a large number of generators. The voltage regulators and governors are limited devices (i.e., there are discontinuities in derivatives). Since implicit methods are not self starting, one has to restart the implicit algorithm each time a discontinuity is encountered. As shown in (x), this problem is solvable, but requires some extra care.

REFERENCE

[x] H. L. Fuller, P. M. Hirsch, and M. B. Lambie, "Variable Integration Step Transient Analysis: VISTA", 1973 PICA Conference Proceedings, pp. 277-284, (1973).

Manuscript received February 5, 1976.

A. Semlyen (University of Toronto, Toronto, Ontario, Canada): Most good models of real systems – and power systems are no exception – suffer from the consequences of stiffness. Therefore, the application of new algorithms for integration of stiff differential equations to the solution of power system dynamics is a useful exercise. The authors have done a commendable job by: (1) reviewing and analyzing the available methods, (2) identifying specific structure (A15) of power system modelling and, (3) developing a stable, large step algorithm for efficient solution. However, because of the mathematical rigor coupled with space limitations, the reader is confronted with increasing hardship and uneasiness, as he progresses towards the final algorithm, but the excellent results displayed in Fig. 4 provide relief and confidence. This discusser is seeking clarification related to his interpretation of equations (A15).

As mentioned by the authors, the stiff equations (A15a) have smaller time constants than the nonstiff equations (A15b). If this separation is significant and the interest is basically in the long duration dynamics, would it be justified to assume that $d\underline{i}/dt = 0$ in (A15a), and obtain $\underline{i} = F(\underline{\delta})$, for the last known $\underline{\delta}$, from the resultant algebraic equations? Then (A15b) can be integrated by a multistep explicit numerical method, using a permissibly large time step. Since, most certainly, such an approach has occurred to the authors, did they find it inadequate for some reason, insufficiently accurate, or is the eigenvalue separation between the two sets of equations not a generally valid assumption? It is conceivable that direct axis rotor transients are slow even compared to mechanical oscillations.

Manuscript received February 6, 1976.

H. W. Dommel (The University of British Columbia, Vancouver, B.C., Canada): The authors clearly demonstrate the numerical stability of implicit integration schemes for generator models with damper windings. The curves of Fig. 4 illustrate very nicely how implicit integration can get through the fast transients in the damper windings with fairly large stepsizes, without running into the smallest time constants barrier of Runge-Kutta methods and other explicit techniques. A permissible stepsize of 0.1s confirms the experience of [9], where only transient time constants and reactances were included and where the hope was expressed that such large stepsizes might still be possible when subtransient effects are included in the model. The authors are to be congratulated for verifying this conjecture now for a whole class of implicit integration schemes.

This discusser agrees that the differential equations (A15) should be retained in split form. A similar approach was used intuitively in [9], but the authors now show that it works because the equations are split into a stiff and nonstiff part. An attempt was made to see how Dahl's angle predictor formula of 1938 with D = 0 (ref. 10 in [9]) compares with Eq. (5). For D = 0, Eq. (5) appears to become implicit when expressions of the form $(1 - \exp(-\gamma))/\gamma$ are set equal to 1.0 for $\gamma = 0$. If this is correct, then some nonzero damping constant should always be included, which is probably more realistic anyhow.

The authors seem to eliminate all non-generator busses, which means that all loads must be represented as constant impedances. This is not always common practice, and it would be interesting to know whether their method could be modified to accommodate "nonimpedance loads". Load models with voltage dependence can improve the accuracy of stability simulations if the models are carefully chosen. On the other hand, errors can be made and probably are made in the definition of voltage-dependent load models in very large system studies, where data is often exchanged among many companies. Therefore, the assumption of constant impedance loads may not be too restrictive for large system studies.

While the Jacobian matrix of Eq. (A15a) or of Eq. (7) is indeed sparse, it is by far less sparse than the admittance matrix used in [9]. In a system with 400 generators, each row of $C(\delta)$ would contain 800 nonzero elements. This, plus the fact that the matrix in Eq. (12) must be retriangularized in every time step, or at least every few time steps, may make the new method less attractive for large systems than that of [9].

Did the authors compare the backward Euler method in Eq. (12) with the trapezoidal rule? This appears to be one of the main differences between the authors' first order method and the method used in [9], realizing of course that the proposed class of algorithms is much more general and in no way restricted to first order schemes. Higher order methods may lose some of their advantages, however, when exciters and governors are included and if limiters in their control circuits are frequently hit.

Manuscript received February 9, 1976.

T. S. Dillon (Monash University Melbourne, Australia): The authors are to be commended for an excellent piece of work in this area. They have partially decoupled the stiff (electrical equations) and nonstiff portions (mech. equations) of the differential equations. They have treated the nonstiff differential equations by a combination of convolution and an Adams-Moulton type method and the stiff differential equations using Gear's method [A,15].

Gear's method is a recognized powerful tool for treating stiff differential equations, the main weakness in its make up is the requirement of the inversion of a Jacobian in the solution of the corrector equation. By exploiting the sparsity properties of the Jacobian in their case, the authors appear to have overcome this weakness. This application of a stiffly stable method to the solution of the stiff differential equations permits the use of large step sizes as the authors state. This recognition and use of stiffly stable methods for the dynamic transients problem, together with the numerical solution of the corrector equation, is an important contribution by the authors to the solution of this problem.

It is worth briefly discussing here the need for the notion of stiff stability referred to by the authors in their paper which was first defined by Gear [A]. Dahlquist [B] has proved that multistep methods that are Absolutely Stable cannot be of order greater than two. Hence a viable notion of numerical stability had to be introduced which was not as restrictive as Absolute Stability but still resulted in numerically meaningful results. Absolute stability implies that the parasitic components $c_v, \underline{i}_p^g$ do not diverge or grow as the solution proceeds. However if the actual solution grows more rapidly than the parasitic components, the situation is acceptable. Furthermore, if the solution tends to zero, the error terms should decrease as rapidly as the solution for it to be meaningful.

These requirements are largely complied with by the definition of stiff stability. For this case, as the authors state, multistep methods of order $\leqslant 6$ have been produced that are stiffly stable.

The main advantage of multistep methods is their use of information available at previous integration points, leading to reduced computation for each step compared to single step methods. However they experience difficulties, if the solution has numerous discontinuities.

The discussors would like the authors' comments on the following points:

(1) Gear [A] in his original paper, suggested the use of variable order, variable step-size in the application of the method. The authors use variable order as a starting method, however it is not clear if they use variable order, variable step-size for error control and computation reduction.

(2) While it is true that their method leads to an increased stepsize compared to the other predictor-corrector schemes, it appears to involve more computation per step because of the solution of the corrector equation (7). A proper comparison of methods in such a situation, should not only be on the basis of step-size but also time of computation per step. Could the authors give figures for this?

Manuscript received February 9, 1976.

(3) Whilst multistep methods perform well on differential equations without discontinuities, they have problems in the presence of discontinuities, because, in general, they are not self-starting. The authors method of using a variable order multistep method for starting, is self-starting. However the low-order steps would show lower accuracy. Have the authors done any comparisons of integration of these equations, with other methods in the presence of discontinuities of the type normally experienced? How does the method compare with single step methods in these cases?

(4) Given the difference in the size of time constants for electrical and mechanical equations, Humpage et al [13] have suggested the use of different step lengths for integration of the equations 15(a) and 15(b). i.e. Could 15(b) be only integrated in multiples of h the step-size used for 15(a). Have the authors investigated such an approach?

REFERENCES

[A] C. W. Gear "The Automatic Integration of Stiff Differential Equations," in Information Processing, 68, ed. A. J. H. Morrell, North Holland Publishing Company, Australia, 187-193, 1969.

[B] G. Dahlquist "A Special Stability Problem for Linear Multistep Methods", B.I.T., 3, 27-43, 1963.

B. Dembart and **K. W. Neves** (Boeing Computer Services, Inc., Seattle, WA): In this paper the authors discuss several aspects of transient stability computation. They point out that in systems where the mechanical swing equations are much slower than the electrical coupling equations, it is quite proper to use different integration methods on the fast and slow components of the system. Furthermore, taking advantage of the exact nature of the swing equation, the authors have introduced a clever and novel integration algorithm that appears to have good accuracy, is stable enough for the slowly varying machine angles, and is explicit (i.e., machine angles at time t_{n+1} can be computed directly from system variables at time t_n which are already known). The fast electrical transients are solved using the backward differentiation formulas that have been selected for use in Gear's algorithm [1].

However, in Gear's algorithm step size and order are controlled to provide highly accurate results. The work required for the high order backward differentiation formulas seems excessive for a fixed step algorithm, or where high accuracy is not required. Furthermore, the stability regions of the high order formulas are not as satisfactory as those of the lower order formulas; especially in light of the fact that the Park's transformations can introduce eigenvalues with large imaginary parts.

Another difficulty associated with the backward difference formulas, without stepsize control, is that if the system is unstable with a large eigenvalue with positive real part associated with the electrical coupling equations, the backward differentiation formulas will not sense the instability, but will give stable results.

Use of the trapezoidal rule with a region of stability corresponding exactly to the left half plane avoids all of these difficulties, and might be a better choice for the electrical coupling equations along with the proposed algorithm for the slow swing equations (particularly for a fixed step algorithm).

In the long term problem the power system controls (mainly turbines) become the slow equations in the system and the swing equations are fast by comparison. The proposed new algorithm does not seem to be suitable for fast transients (see [2] where this method is compared to a similar but different one for the 2-step case).

REFERENCES

[1] Gear, C., "The Automatic Integration of Stiff Ordinary Differential Equations", IFIPS Conference Proceedings, 1968.

[2] Semlyen, A., "Calculation of Transients in a System of Algebraicaly connected Dynamic Components Using a New Numerical Algorithm", IEEE Trans. Circuits and Systems, Vol. CAS-22, pp. 622-627, July 1975.

Manuscript received February 13, 1976.

William W. Price and **Richard P. Schulz** (General Electric Company, Schenectady, New York): The authors are to be commended for moving forward in the area of the integration of power system dynamic equations. Particularly commendable is their recognition of the particular structure of the power system equation solutions rather than rigidly applying techniques developed in other disciplines.

The discussors are uncertain as to the intended application of this technique. The title, text and generator model, including rate-of-change of flux terms, would imply broad application beyond transient stability. With these terms included, the 60 Hz and other high frequency oscillations will appear in the solution and a small time step will be required to accurately reproduce these phenomena. The corresponding terms from *all* of the network elements should also be included in this case. On the other hand, if power-angle swings are the major concern, then elimination of the rate-of-change of flux terms in the machine equations would permit longer time steps for the conventional integration methods. Therefore, we believe the comparison of methods in this paper is on a somewhat unfair basis.

The discussors have the opinion that a very simple integration algorithm, such as the Euler or modifications thereof, may require more time steps, but by its simplicity minimizes total computation time. We have developed both transient stability[1] and higher frequency transients[2] programs utilizing such an algorithm. While we are continually on the lookout for breakthroughs in this area and believe they may come, we do not believe any algorithms developed to date have demonstrated significant overall simulation cost improvements with similar solution accuracy.

We would therefore urge the authors and others developing new algorithms to make comparisons, not only on the basis of reduction in the number of time steps or on the basis of comparison to high order Runge-Kutta algorithms, but on the basis of total computation time compared to a very simple algorithm. If the authors could provide such a comparison, it would be greatly appreciated.

REFERENCES

1. FACE Multimachine Power System Simulation, D. N. Ewart, R. P. Schulz, 1969 PICA Conference.
2. MANTRAP – Machine and Network Transients Program, K. Carlsen, E. H. Lenfest, J. J. LaForest, 1975 PICA Conference, pp. 144-151.

E. Paulsson and **J. Bubenko** (The Royal Institute of Technology, Stockholm, Sweden): The authors are to be commended for an interesting paper, which contributes to the literature on the subject.

Authors' comments on the following will be appreciated.

1. The results obtained by using the proposed integration techniques show that large stepsizes could be used after the initial high frequency transients have been damped out. This is also true for all stiffly stable methods. We have also noticed that no control systems were included in the power system model. By adding those to the system a number of limits will be introduced, and each time a state variable reaches such a limit a discontinuity in the solution will occur. In other words the solution is not always differentiable and all points where limits are reached should be treated as new starting points resulting in a first order approximation. Besides, due to discontinuities, transients associated with small time constants do not necessarily damp out. How do these phenomena affect the choice of step-size when using the proposed technique?

2. How will control systems and more general load-models affect the structural properties of the mathematical model? Is it still possible to use the same solution algorithm?

Manuscript received February 17, 1976.

Brian Stott (University of Waterloo, Ontario, Canada): The paper is a welcome addition to the growing volume of literature that emphasises the advantages of stiffly-stable implicit integration methods for power system dynamic response calculations. My comments relate generally to these classes of methods, and are based on a long period of research during which we have been comparing a wide range of modern methods, including Laplace/Pade, Gear-type and other multistep variable-order

Manuscript received February 17, 1976.

Manuscript received February 20, 1976.

implicit schemes. They encompass iterative and noniterative techniques and approaches where the differential and algebraic equations are solved alternately or together as one algebraized set. The multi-machine test problems have included fast limiting excitation controls with saturation, non-impedance bus loads and induction motor models.

Stable integration methods are of considerable value in promoting overall computing efficiency and reliability. They inhibit error growth from system stiffness and all other sources. We find that once a sufficiently stable integration method (i.e. any one of the better implicit schemes) has been adopted, stiffness presents relatively little problem. Apart from the question of discontinuities that arise mostly during the "transient" period and impose practical limits on step sizes in that region, the main problem is system nonlinearity.

In approaching the objective of minimizing (computation per step)/(step size), the general problem can be attacked by using (a) very stable integration methods, (b) high-accuracy integration formulas, probably using previous values, (c) system model formulations that retain as much linearity as possible, (d) efficient computational techniques for solving the differential and algebraic equations either together or separately, and (e) computational techniques that minimize the generation of errors, particularly those due to nonlinearity (including interfacing).

We find that point (e) is a very important one. When exploiting quasi-linear structure in pursuance of points (a) – (d), noniterative methods tend to incur high linearization errors that turn out to be the main limiting factor in extending step sizes. I think that the paper has not entirely got to grips with this problem because the system modelled does not contain excitation control, non-impedance loads, saturation, etc. Nor is it easy to predict how comparatively effective the paper's method would be if they were included.

So far, we have not obtained dramatic improvements over methods based on Reference 9, which employs iteration and fulfills points (a), (c) and (e) very well and point (d) fairly well. Indeed, for shorter-term dynamics (a few seconds) they seem to be as good as anything that we have tried, and have a number of implementational advantages. In addition, iteration is a good measure of problem nonlinearity and provides a very practical criterion for step-size adjustment.

The authors' method appears to use a nonsparse reduced network matrix. Can this be competitive for large systems, and how can non-impedance loads be incorporated?

Work along the general lines described in the paper is expected to produce worthwhile economies for longer-term dynamics. The more-cumbersome program structures required for implicit methods are perhaps their only disadvantage relative to conventional explicit methods, but no doubt the industry will accept this necessary evil just as it has accepted sparse matrix methods.

Demos Gelopulos and **Denny Brown** (Arizona State University, Tempe, AZ): We would like to make one clarification and pose two questions regarding this interesting paper.

(1) The reference to "Converti, et al", (ref. 14) incorrectly assumes that high dimensional state transition matrices are computed. The highest subsystem dimension is eight (8) and it is very simple to compute the transition matrices needed. Since these are only computed at discontinuities and not at each time step, the fraction of the computational task devoted to calculation of state transition matrices is quite insignificant.

(2) For Arizona Public Service Company's Transient Stability needs we have found (see "Comparison of System Representation for Reliable Stability Assessment", V. Converti, G. Steinbrenner, D. Gelopulos & M. Housley, Midwest Power Symposium Conference Record, October, 1974; and "Dynamic Modelling of Loads in the Stability Studies", M. H. Kent, W. R. Schmus, E. A. McCrackin & L. M. Wheeler, IEEE Trans., May, 1969) that the system response is sensitive to load models and that constant impedance loads are not satisfactory. Excluding integration schemes such as explicit P-C methods or Runga-Kutta, the minimum time step is limited by the interface between equation (1a) and (1b). It is difficult to handle this interface accurately if nonlinear load models are used.

What would be required to allow the use of constant current and constant power load models in this new method?

(3) The oscillations in the example plots appear to occur at a very high frequency (tens of hertz). The authors assume that the network is in a pseudo steady state as is customary in Transient Stability problems.

Would the authors please comment on the reason for using an example with such a fast response?

Manuscript received August 13, 1976.

George Gross and **Arthur R. Bergen**: We appreciate very much the interest of the discussers in this work. We thank them for their comments and welcome the opportunity provided to clarify some of the points in our paper.

Our principal objective in this paper was to explore the structural properties of the power system dynamical equations and to make use of these properties to develop a class of efficient integration methods for simulating power system dynamical response. For the model used in this paper we have derived a very effective noniterative solution scheme which permits the use of relatively large stepsizes. While we are encouraged by our progress thus far, we are aware, as are the discussors, of many topics requiring further study in this area.

An important topic is the numerical treatment of the associated control equipment (governors, exciters and power system stabilizers) dynamics – the most difficult aspect of the numerical integration problem according to Dr. Hirsch. Certainly, as most of the discussors point out, the presence of the equations of motion of these dynamics cause complications; in particular, the d.e.s. become discontinuous whenever the limits are reached. The problem of dealing with discontinuities in the numerical integration process has received very little attention in the field of numerical analysis in general and power system area in particular. If in the course of simulation the limits are hit very frequently, this would, admittedly, make multistep methods less attractive since at each such instance the solution of a *new initial value problem* is required. Also, as pointed out by Drs. Paulsson and Bubenko, these discontinuities may effect the choice of stepsize. In order to represent the discontinuous waveforms accurately, it may become necessary to use smaller stepsizes for the control circuits and some multiple of this stepsize for the remainder of the system. The study of the integration of equations arising from the controls of the power system is a logical extension of the work reported here and we hope to present some new results in the near future.

Profs. Dommel and Stott note correctly that the reduced network matrix used in our scheme is generally nonsparse. This, then has the effect to make the Jacobian matrix in Step 3 of the Algorithm (eq. (12)) less sparse than the matrices usually encountered in other power system problems. We have formulated the algorithm in terms of the reduced impedance matrix in order to give a more "closed form" solution to the problem. This is not necessary since we can restate the formulation so that a very sparse matrix is used. In this way, we can also accommodate nonimpedance loads, a matter of concern to many of the discussers. The following is a brief sketch of the modification required.

We consider a power system with n generator nodes 1, 2, . . . , n and q nongenerator (or load) nodes which we denote as n+1, n+2, . . . , n+q. Let $\hat{V}(\hat{I})$ denote the vector of the real and imaginary parts of the bus voltages (current injections). Note that $V=(V^N,V^L)$ $(I=(I^N,I^L))$ where $V^N(I^N)$ is as defined in (A5) and $V^L(I^L)$ is the component of $\hat{V}(\hat{I})$ at the q nongenerator nodes. The network behavior is described by:

$$\underline{0}=\underline{g}(\hat{\underline{V}},\hat{\underline{I}}) \tag{D1}$$

Here g is a nonlinear function that represents the nonlinear loads and the network voltage and current constraints. (If all loads are constant impedances, then g becomes the linear equation

$$\underline{0} = \hat{\underline{I}} - \hat{\underline{Y}}\hat{\underline{V}} \quad \text{or} \quad \underline{0} = \hat{\underline{V}} - \hat{\underline{Z}}\hat{\underline{I}} \tag{D1'}$$

where $\hat{Y}(\hat{Z})$ is the network admittance (bus impedance) matrix. Upon eliminating the equations associated with the q nongenerator nodes we obtain the reduced network representation of (A5)). Turning now to the equations describing the electrical side of the generators we have from (A9), (A10) and (A13) that

$$\frac{d\underline{i}_\nu}{dt} = \underline{A}_\nu \underline{i}_\nu + \underline{b}_\nu + \underline{F}_\nu \underline{v}_\nu^m \qquad \nu = 1, 2, \ldots, n \tag{D2}$$

where

$$\underline{F}_\nu = \underline{L}_\nu^{-1} \underline{E}_\nu \qquad \nu = 1, 2, \ldots, n \tag{D3}$$

Then for the system of n generators we obtain

$$\frac{d\underline{i}(t)}{dt} = \underline{A}\,\underline{i}(t) + \underline{b} + \underline{F}\,\underline{v}^m(t) \tag{D4}$$

where, $F = \text{diag}\,[F_\upsilon: \upsilon = 1, 2, \ldots, n]$. The machine rotor based variables v_υ^m and i_υ^m are related to the network based variables V_υ^N and and I_υ^N via the transformations $T(\delta_\upsilon)$, $\upsilon = 1, 2, \ldots, n$ defined in (A6). Thus

$$\underline{I}^N = \underline{T}^{-1}(\underline{\delta})\underline{i}^m = \underline{T}^{-1}(\underline{\delta})\underline{E}\,\underline{i} \tag{D5}$$

and

Manuscript received September 2, 1976.

$$\underline{v}^N = \underline{T}^{-1}(\delta)\underline{v}^m$$

where $T(\delta) = \text{diag}\,[T(\delta_\upsilon)\colon \upsilon = 1, 2, \ldots, n]$. Then the electrical system is described by (D1), (D4), and (D5). The modification of the Algorithm for this more general formulation occurs in Step 3. At this step the differential equation (D4) must be solved together with the algebraic system (D1). Using backward differentiation to solve (D4), we obtain

$$\left[\underline{U} - h\mu_{k,0}\underline{A}\right] \underline{i}^{\ell+1} - \left[h\mu_{k,0}\underline{F}\right](\tilde{\underline{v}}^m)^{\ell+1} = h\mu_{k,0}\underline{b} + \sum_{\nu=1}^{k} \mu_{k,\nu}\underline{i}^{\ell+1-\nu} \qquad \text{(D6)}$$

where we use $(\tilde{v}^m)^{\ell+1}$ to denote the computed value of v^m at time $t^{\ell+1}$. In addition we must solve

$$\underline{0} = \underline{g}\left(\left[\underline{T}^{-1}(\underline{s}^{\ell+1})(\underline{\tilde{v}}^m)^{\ell+1}\right], (\underline{\tilde{v}}^L)^{\ell+1}, \left[\underline{T}^{-1}(\underline{s}^{\ell+1})\,\underline{E}\,\underline{i}^{\ell+1}\right], (\underline{\tilde{i}}^L)^{\ell+1}\right) \qquad \text{(D7)}$$

We can solve for $\underline{i}^{\ell+1}, (\tilde{\underline{v}}^m)^{\ell+1}$ in (D6)–(D7) using Newton Raphson. Note that (D6) is linear in $\underline{i}^{\ell+1}$ and $(\underline{v}^m)^{\ell+1}$ with coefficient matrices $U - h\mu_{k,0}A$ which is block diagonal and $-h\mu_{k,0}F$ which is very sparse. Furthermore, the partials of (D6) have the customary sparsity associated with the network equations [9]. For the linear case (D1′) the solution is obtained in one step requiring no iterations. A possible alternative scheme for a nonlinear g is the iteration: i) solve for $\underline{i}^{\ell+1}$ in (D6) using the value $(\tilde{\underline{v}}^m)^\ell$; ii) use this value of $\underline{i}^{\ell+1}$ to solve for $(\tilde{\underline{v}}^m)^{\ell+1}$ in (D7) using Newton's method, and iii) if necessary return to (i). For both schemes, the sparsity can be efficiently exploited. It follows then that nonlinear loads can be treated within the framework of our Algorithm in an effective manner.

We concur with Prof. Dommel that a nonzero damping constant should always be included to more realistically model the system. On the other hand, we disagree with his observation that the eq. (5) becomes implicit for $D_i = 0$, $i = 1, 2, \ldots, n$. Indeed, even for this case $s^{\ell+1}$ can be computed explicitly. We establish this by using the relation:

$$\lim_{\xi \to 0} \frac{1 - \exp(-\xi)}{\xi} = 1 \qquad \text{(D8)}$$

For the case $D_i = 0$ we have that (5) reduces to

$$\underline{s}^{\ell+1} = \underline{s}^{\ell} + h\underline{u}^{\ell} + \sum_{\nu=0}^{k} \sigma_{k,\nu}(\nu h)\underline{f}(\underline{i}^{\ell+1-\nu}) \qquad (5')$$

Note that the first term in the summation vanishes since $\upsilon = 0$. Consequently, we need to sum only from $\upsilon = 1$ as for the case $D_i \neq 0$. Thus we have that using (5′) we compute $s^{\ell+1}$ explicitly.

Several of the discussers have concentrated their remarks on the computational aspects of numerical integration methods in general and the proposed schemes in particular. We wish to point out in answering these comments that the program we used to test the new schemes is not a very sophisticated one: for example, there is no feature to allow for automatic stepsize and order control, no optimal ordering or sparsity oriented programming is used, etc. Messrs. Dillon, Dembart and Neves have commented on the use of variable order and variable stepsize to integrate the d.e.s. In response, then, to Dr. Dillon's question we have tested the proposed schemes using fixed stepsize and fixed order. Gear's backward differentiation based integration method, as other multivalue/multistep schemes, is usually implemented in production-level programs with techniques for changing step and changing order as necessary. In addition to ensuring that the equations are integrated with sufficient accuracy, for a stiffly stable method such techniques ensure that the region R_{II} of Fig. 1 lies outside the region of absolute stability of the method, i.e., within the closed contour of Fig. 1 corresponding to the order of the method. Recently, some efficient schemes which reduce the additional overhead incurred by such techniques have been proposed [D1], [D2].

Messrs. Price and Schultz strongly advocate the use of the forward Euler integration method or its modification. While we agree with the discussers that such a scheme is very simple to implement and requires only small amount of computation per stepsize, we are less convinced of its suitability for integrating the stiff d.e.s. arising in power system dynamics. As is well known, the forward Euler method belongs to the class of schemes which suffer from the shortest time constant barrier – its region of absolute stability in the complex plane is the unit circle

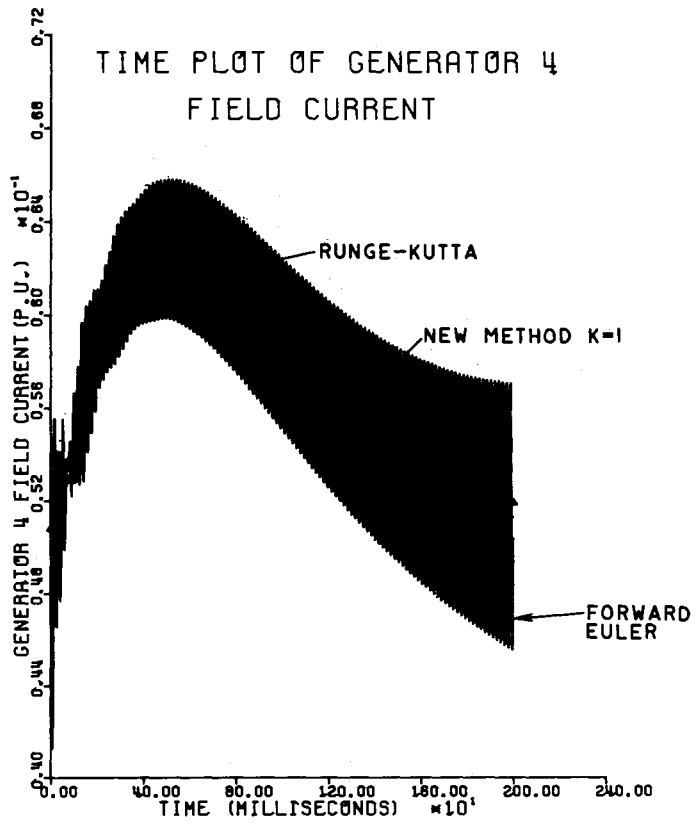


Fig. A
Part a

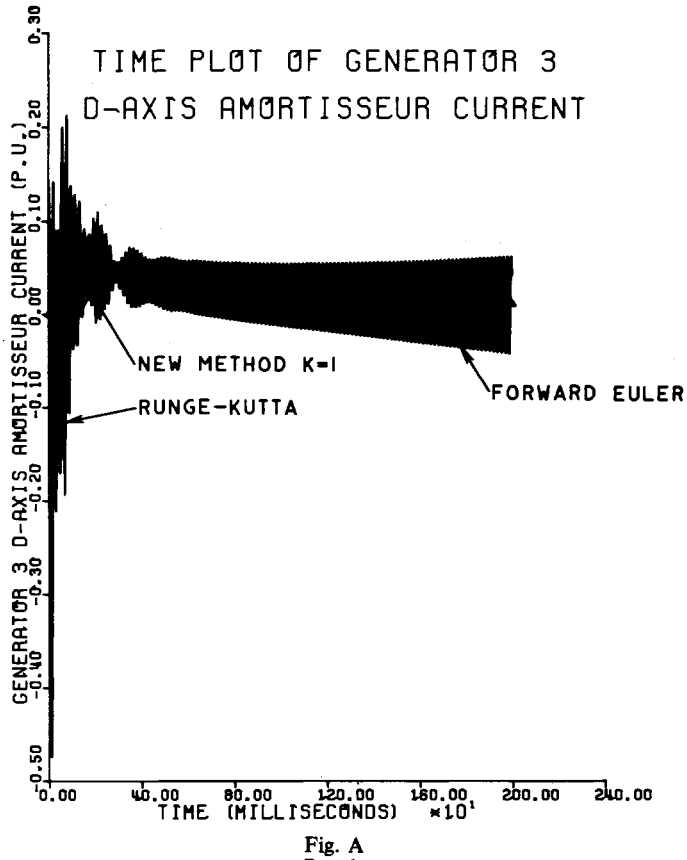


Fig. A
Part b

centered at -1 + j0 (cf. Fig. 2). Moreover, the small stepsize imposed by numerical stability considerations requires, as the discussers state, a large number of steps, which in turn, may cause problems with the roundoff error. We have used forward Euler to integrate the d.e.s. of the example in the paper and found that for numerical stability we required stepsizes less than 10^{-4} sec. The time plots in Fig. A were obtained using forward Euler with a stepsize of 10^{-4} sec.; the corresponding plots using Runge Kutta with a 10^{-4} sec. stepsize and the new scheme with k = 1 and a 10^{-1} sec. stepsize are given for comparison purposes. We recognize the fact that the authors use forward Euler to simulate high frequency transients with a very small stepsize and for transient stability with a larger stepsize using a less detailed machine representation and consequently do not get into the numerically unstable situation illustrated in Fig. A.

Messrs. Dillon, Price, and Schulz have requested some data on the computation times for the new schemes. We have provided this in Table A for the example of the paper. The times given for the new schemes have to be considered in light of the fact that most of the time in an iteration is spent in actually inverting a 20x20 matrix using a canned inversion routine rather than a sparsity based one.

TABLE A

Method	Stepsize h (sec.)	Number of Steps	Computation Time (10^{-2} sec.)	Time/Step (10^{-2} sec.)
new; k = 1	0.1	20	52	2.6000
new; k = 2	0.1	19	50	2.6315
new; k = 3	0.1	18	48	2.6666
new; k = 4	0.1	17	45	2.6470
Euler	0.001	2000	1154	0.577
Runge-Kutta	0.001	2000	4681	2.340

Messrs. Dembart and Neves have cited some of the disadvantages associated with the higher order backward differentiation schemes. While, when high accuracy is desired, one can put up with the additional computational effort for, and/or the somewhat smaller region of absolute stability of the higher order schemes, we believe that the factor that would discourage their use to a much larger extent is the frequent occurrence of discontinuities (see above). These disadvantages lead the discussers to recommend the use of the modified k = 1 scheme mentioned briefly in the paper. In this scheme the backward Euler scheme in Step 3 of the Algorithm is replaced by the trapezoidal rule. We wish to point out that using this modified scheme is computationally not equivalent to application of the trapezoidal method of [9], as claimed by Dr. Dommel. For this modified scheme we use the trapezoidal rule for both numerical quadrature to approximate the integrals in (4) and (8) and numerical differentiation to approximate the derivative in (A15a). On the other hand the trapezoidal rule in [9] is used strictly for the latter purpose. Both schemes are second order methods. It is interesting to note that for the model used here the modified scheme computes the values of the state variables at the new time step requiring no iterations as opposed to the trapezoidal rule with which iterations would be required as (A15) is a set of nonlinear equations in $(i, \delta, \omega)^T$. In fact we used exactly this scheme to implement the simulation of a detailed synchronous machine model including the shaft torsional dynamics, into the BPA Transients Program. We have obtained very satisfactory performance with the modified k = 1 scheme for this problem.

Messrs. Price and Schulz inquire about the intended scope of the proposed scheme. In the previous paragraph we have indicated that we have applied the modified k = 1 scheme for simulating higher frequency transients. We believe that the schemes will be useful for transient stability simulation once exciter dynamics are included. We expect, however, that the schemes will prove of greatest use for the simulation of long term dynamics. For this case the inclusion of the slow governor and even slower boiler dynamics introduces stiffness into the mechanical equations which will be possible to exploit by schemes similar to the ones proposed here. Prof. Semlyen raises a question concerning the validity of the assumption $di/dt = 0$ in (A15a) for the representation of longer duration dynamics. If this assumption were made, we would obtain essentially the classical swing equation model of a multimachine power system. This model is generally deemed invalid for studies of longer duration.

Messrs. Gelopulos and Brown inquire about the choice of the numerical example for our paper. We used this particular test system to validate the class of proposed algorithms because it gives rise to a set of stiff differential equations. The numerical results show that our schemes deal effectively with the wide range of time constants and frequencies that are present in this sample system. The clarification provided by the discussers concerning the reference to [14] is appreciated.

In closing, we are pleased by the wide interest in this area as evidenced by the discussions submitted and express the hope that some of the topics will benefit from further study and research in the near future.

REFERENCES

[D1] R. K. Brayton et al., "A new efficient algorithm for solving differential–algebraic systems using implicit backward differentiation formulas," Proc. IEEE, Vol. 60, pp. 98-108, January 1972.

[D2] W. M. G. van Bokhoven, "Linear implicit differentiation formulas of variable step and order," IEEE Trans. Circuits System, Vol. CAS-22, February 1975.