Pergamon

Mathl. Comput. Modelling Vol. 19, No. 12, pp. 67–84, 1994

0895-7177/94 $7.00 + 0.00

0895-7177(94)00082-4

The Waveform Relaxation Method for Systems of Differential/Algebraic Equations

M. L. CROW
Department of Electrical Engineering
University of Missouri-Rolla, Rolla, MO 65401, U.S.A.

M. D. ILIĆ
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.

(Received and accepted October 1993)

Abstract—This paper reports efforts towards establishing a parallel numerical algorithm known as Waveform Relaxation (WR) for simulating large systems of differential/algebraic equations. The WR algorithm was established as a relaxation based iterative method for the numerical integration of systems of ODEs over a finite time interval. In the WR approach, the system is broken into subsystems which are solved independently, with each subsystem using the previous iterate waveform as "guesses" about the behavior of the state variables in other subsystems. Waveforms are then exchanged between subsystems, and the subsystems are then resolved repeatedly with this improved information about the other subsystems until convergence is achieved.

In this paper, a WR algorithm is introduced for the simulation of generalized high-index DAE systems. As with ODEs, DAE systems often exhibit a multirate behavior in which the states vary at differing speeds. This can be exploited by partitioning the system into subsystems as in the WR for ODEs. One additional benefit of partitioning the DAE system into subsystems is that some of the resulting subsystems may be of lower index and, therefore, do not suffer from the numerical complications that high-index systems do. These lower index subsystems may therefore be solved by less specialized simulations. This increases the efficiency of the simulation since only a portion of the problem must be solved with specially tailored code. In addition, this paper established solvability requirements and convergence theorems for varying index DAE systems for WR simulation.

Keywords—Waveform relaxation, Differential/algebraic equations (DAEs), Initial value problems, Parallel processing.

1. INTRODUCTION

A differential/algebraic system may be modeled in the form

$$0 = \hat{F}(\dot{y}, y, t), \tag{1}$$

where $\frac{\partial \hat{F}}{\partial \dot{y}}$ is singular and $\frac{\partial \hat{F}}{\partial y}$ may or may not be singular. Systems of differential/algebraic equations (DAEs) of this type arise in connection with power systems [1,2], singular perturbation theory [3], control theory [4], circuit simulations [5], robot dynamics [6], and many other applications in the fields of mechanical and chemical engineering, economics, and physics. Only recently has concerted effort been put forth to find methods to numerically solve these systems [4,5,7–15]. Previously, systems of DAEs were frequently restated as ODEs, often with considerable difficulty or by destroying the structure of the problem [12] (i.e., the resulting variables often no longer represent physical quantities, or the inherent sparsity of the system is destroyed), but as DAE systems arise more and more frequently, it has become necessary to develop numerical methods

for solving these systems distinct from the traditional methods for ODEs. If $\frac{\partial \hat{F}}{\partial y}$ is invertible, this is a sufficient, but not necessary, condition for the system of (1) to be index one. If $\frac{\partial \hat{F}}{\partial y}$ is noninvertible, the system is said to be of *higher* index [13]. Difficulties in using ODE methods for solving DAE systems occur when the systems have index greater than or equal to two.

Standard circuit simulators use direct methods to discretize the system by standard stable implicit integration methods. Modified Newton methods are then employed to iteratively solve the resulting algebraic equations, and a sparse Gaussian elimination is utilized to solve the systems of linear equations produced by the Newton method [16]. The direct method may become inefficient for very large, dynamic systems. This is because the matrix solution time of the linear algebraic equations grows super linearly with the size of the problem, thus swamping all other steps of the integration method. The direct methods are also inefficient for systems with states which are varying with considerably different rates. Direct application of a discretizing integration method forces all of the states to be discretized identically and with sufficient fineness such that the fastest changing state can be accurately reproduced. If it were possible to divide the system into several subsystems, each of which were changing at individual rates, then it would be possible to integrate each subsystem with the largest possible time step which would accurately reflect the behavior of the subsystem.

In addition, if it were possible to divide a higher index system into subsystems, many of the subsystems may possibly have a lower index, making it possible to solve these subsystems with the usual numerical methods, with little or no additional precautionary measures, while those of higher index may be integrated using methods specifically tailored for high index systems.

One method which overcomes all of the above drawbacks is the *waveform relaxation* (WR) algorithm. The WR algorithm was introduced as an iterative method for the numerical integration of the system of ordinary differential equations over a finite time interval [17]. It is based on the Gauss-Seidel and Gauss-Jacobi relaxation methods [18] used for solving large systems of algebraic equations. In the WR approach, the system is broken into subsystems which are solved independently, with each subsystem using the previous iterate waveforms as "guesses" about the behavior of the state variables in other subsystems. Waveforms are then exchanged between subsystems, and the subsystems are then resolved with improved information about the other subsystems. This process is repeated until convergence is achieved. The WR algorithm was first applied to index one DAEs in [19] for the simulation of VLSI circuits. This algorithm for ODEs was further explored in [20]. This paper endeavors to generalize the WR algorithm to encompass a broader spectrum of DAE systems, namely those of higher index.

Because the Gauss-Jacobi WR algorithm is inherently parallel in nature, it is well suited for implementation on various parallel processors. The suitability of the Gauss-Siedel WR algorithm is strongly dependent on the implementation.

2. THE WAVEFORM RELAXATION ALGORITHM FOR SYSTEMS OF DIFFERENTIAL/ALGEBRAIC EQUATIONS

An important subclass of nonlinear DAEs has the form

$$\dot{\mathbf{x}}(t) = F(\mathbf{x}, \mathbf{y}, t), \qquad \mathbf{x}(0) = \mathbf{x}_0, \tag{2}$$

$$0 = G(\mathbf{x}, \mathbf{y}, t), \tag{3}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{y}(t) \in \mathbb{R}^m$, $F : \mathbb{R}^{n+m+1} \to \mathbb{R}^n$, and $G : \mathbb{R}^{n+m+1} \to \mathbb{R}^m$. This type of system is known as a *semi-explicit* DAE. For a nonlinear system of this type, a local and global index can be defined [7]. The local index is the index on the linear constant coefficient system that results from linearizing a nonlinear system at a given fixed time. The global index is the number of times the nonlinear DAE system must be differentiated to obtain a system of ODEs [13]. In summary, the index might be considered to be a measure of the singularity of the system. In this

work, only systems where the local and global indices are the same are considered. For a more general discussion, see [9].

The WR algorithm is a means of solving a large system of nonlinear semi-explicit DAEs. The basic method proposed here is to partition the system into subsystems in which tightly coupled state and nonstate variables are grouped together. In particular, the system is decomposed into r subsystems as

$$\dot{\mathbf{x}}_1(t) = F_1(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_r, \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_r, t), \quad \mathbf{x}_1(0) = \mathbf{x}_{10}, \tag{4}$$

$$0 = G_1(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_r, \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_r, t), \quad \mathbf{y}_1(0) = \mathbf{y}_{10}, \tag{5}$$

$$\vdots \qquad\qquad \vdots$$

$$\dot{\mathbf{x}}_r(t) = F_r(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_r, \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_r, t), \quad \mathbf{x}_r(0) = \mathbf{x}_{r0}, \tag{6}$$

$$0 = G_r(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_r, \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_r, t), \quad \mathbf{y}_r(0) = \mathbf{y}_{r0}, \tag{7}$$

where $\mathbf{x}_i \in \mathbb{R}^{n_i}$, $\mathbf{y}_i \in \mathbb{R}^{m_i}$, $\sum_{i=1}^{r} n_i = n$, $\sum_{i=1}^{r} m_i = m$, $F_i : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{n_i}$, and $G_i : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{m_i}$. The Gauss-Jacobi WR algorithm for solving (4)–(7) is given in Algorithm 2.1.

ALGORITHM 2.1. *The Gauss-Jacobi WR Algorithm for Semi-Explicit DAEs.*

$k \leftarrow 0$;
Guess some $\mathbf{x}^{k+1}(t)$ *such that* $\mathbf{x}^{k+1}(0) = \mathbf{x}(0)$;
Guess some $\mathbf{y}^{k+1}(t)$ *such that* $\mathbf{y}^{k+1}(0) = \mathbf{y}(0)$;
repeat{
  $k \leftarrow k + 1$;
  foreach $(i \in \{1, \ldots, r\})$ *solve on* $[0, T]$

$$\dot{\mathbf{x}}_i^{k+1} = F_i(\mathbf{x}_1^k, \ldots, \mathbf{x}_i^{k+1}, \ldots, \mathbf{x}_r^k, \mathbf{y}_1^k, \ldots, \mathbf{y}_i^{k+1}, \mathbf{y}_r^k, t) \quad \mathbf{x}_i^{k+1}(0) = \mathbf{x}_i(0)$$

$$0 = G_i(\mathbf{x}_1^k, \ldots, \mathbf{x}_i^{k+1}, \ldots, \mathbf{x}_r^k, \mathbf{y}_1^k, \ldots, \mathbf{y}_i^{k+1}, \mathbf{y}_r^k, t) \quad \mathbf{y}_i^{k+1}(0) = \mathbf{y}_i(0)$$

}**until** $(\|\mathbf{x}^{k+1} - \mathbf{x}^k\| \le \epsilon_x$ *and* $\|\mathbf{y}^{k+1} - \mathbf{y}^k\| \le \epsilon_y)$; *where* ϵ_x *and* ϵ_y *are small positive values.*

The WR algorithm is iterative in nature, with the previous iterate waveforms of both differential and algebraic variables acting as inputs to the subsystem currently being solved.

2.1. The Solvability of the WR Algorithm for Systems of DAEs

The notion of solvability for DAE systems [7] can be extended to DAE systems with waveform relaxation applied.

THEOREM 2.1. *If the system*[1]

$$\mathbf{E}\,\dot{\mathbf{y}}(t) = \mathbf{A}\,\mathbf{y}(t) + \mathbf{B}\,\mathbf{U} \tag{8}$$

is solvable, then the waveform relaxation formulation is solvable if and only if all the subsystems are solvable.[2]

There are two important observations to make from this theorem. The first observation is that this theorem does not imply convergence of the WR method. It only states that for a solvable system, given any continuous input vector $\mathbf{y}^k$, then there exists a unique output vector $\mathbf{y}^{k+1}$. It does not guarantee anything about the relationship of $\mathbf{y}^k$ to $\mathbf{y}^{k+1}$. The second observation is that the solvability of the WR method formulation depends on the subsystems chosen. If two possible

[1] $\mathbf{E}$ may or may not be singular.
[2] The proof of this theorem is straightforward, but lengthy and is omitted for brevity. The proof may be found in [21].

ways of partitioning the total system into subsystems are chosen, one may be solvable while the other is not. Thus, the solvability of the entire system does not guarantee the solvability of the partitioned system.

In some cases, applying WR may destroy the solvability of the system. As an example of a problem where the system is solvable, but the subsystems are not, consider the index three simple pendulum problem

$$\ddot{x} = L\,x, \tag{9}$$

$$\ddot{y} = L\,y + g, \tag{10}$$

$$0 = \frac{1}{2}\left(x^2 + y^2 + \ell^2\right), \tag{11}$$

where x and y are the planar coordinates of an infinitesimal ball of unit mass at one end of a bar of length ℓ, L is proportional to the force in the bar, and g is the gravity constant. This system may be written equivalently in standard explicit DAE form

$$\dot{v} = L\,x, \tag{12}$$

$$\dot{x} = v, \tag{13}$$

$$\dot{u} = L\,y + g, \tag{14}$$

$$\dot{y} = u, \tag{15}$$

$$0 = \frac{1}{2}\left(x^2 + y^2 + \ell^2\right). \tag{16}$$

The equilibrium point for this system is $(0, 0, 0, -\ell, g/\ell)$. The solvability of this system is determined by considering [5,7]

$$\det\left(\mathbf{J}(v, x, u, y, L) - \lambda\,\mathbf{E}\right).$$

The determinant is not identically zero for any points of interest; therefore, this system is solvable. After applying WR, the partitioned system becomes

$$\dot{v}^{k+1} = L^{k+1}\,x^{k+1}, \tag{17}$$

$$\dot{x}^{k+1} = v^{k+1}, \tag{18}$$

$$0 = \frac{1}{2}\left((x^{k+1})^2 + (y^k)^2 + \ell^2\right), \tag{19}$$

$$\dot{u}^{k+1} = L^k\,y^{k+1} + g, \tag{20}$$

$$\dot{y}^{k+1} = u^{k+1}. \tag{21}$$

The respective Jacobians are

$$\mathbf{J}_x = \begin{bmatrix} 0 & L^{k+1} & x^{k+1} \\ 1 & 0 & 0 \\ 0 & x^{k+1} & 0 \end{bmatrix}, \qquad \mathbf{J}_y = \begin{bmatrix} 0 & L^k \\ 1 & 0 \end{bmatrix}.$$

The subsystem of the $\{x, v, L\}$ variables is not solvable as x^{k+1} approaches 0, which is also the equilibrium point, since

$$\det\left(\mathbf{J}_x - \lambda\,\mathbf{E}_x\right) = \det\begin{bmatrix} -\lambda & L^{k+1} & x^{k+1} \\ 1 & -\lambda & 0 \\ 0 & x^{k+1} & 0 \end{bmatrix} \to 0,$$

as $x^{k+1} \to x_{\text{eq}} = 0$. Thus, the waveform relaxation algorithm cannot be used to solve this problem unless the whole system is taken as one partition, which defeats the purpose of the algorithm. The interdependence of system partitioning and rate of convergence of the waveform iterates is discussed in detail for ODEs in [22].

2.2. Convergence of the WR Algorithm for General Index One Systems

Most DAEs may generally be modeled as

$$0 = F_1(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t), \tag{22}$$

$$0 = F_2(\mathbf{x}, \mathbf{y}, t), \tag{23}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{y}(t) \in \mathbb{R}^m$ and $\left[\frac{\partial F_1}{\partial \dot{\mathbf{x}}}\right]$ is nonsingular. This is known as a *general form* DAE. Integrated MOS circuits containing pass transistors fall into this category and can be written as follows:

$$0 = \mathbf{C}(\mathbf{v}, \mathbf{u})\,\dot{\mathbf{v}}(t) + \mathbf{q}(\mathbf{v}, \mathbf{z}, \mathbf{u}), \qquad \mathbf{v}(0) = \mathbf{V}, \tag{24}$$

$$0 = \mathbf{z}(t) - \mathbf{g}(\mathbf{v}, \mathbf{u}), \tag{25}$$

where $\mathbf{C}$ is a symmetric diagonally dominant matrix-value function in which $C_{ij}(\mathbf{v}, \mathbf{u})$, for $i \neq j$, is the total floating capacitance between nodes i and j, $\mathbf{v}$ is the vector of unknown voltages, $\mathbf{u}$ is the vector of all inputs and their derivatives, and $\mathbf{z}$ is the vector of drain currents of the pass transistors.

When the WR method is applied to index one systems of the form of equations (22) and (23) this gives rise to the canonical form[3]

$$\dot{\mathbf{w}}^{k+1} = \hat{F}_1(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k), \tag{26}$$

$$\mathbf{z}^{k+1} = \hat{F}_2(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k), \tag{27}$$

where $\mathbf{w}(t) \in \mathbb{R}^n$ and is possibly different from $\mathbf{x}(t)$, $\hat{F}_1$ and $\hat{F}_2$ are continuous mapping functions of appropriate dimensions, which are possibly (usually) different from F_1 and F_2, and the algebraic variable $\mathbf{z}(t) \in \mathbb{R}^m$ is possibly (usually) different from $\mathbf{y}(t)$.

This form of WR applied to DAEs was explored in [19]. The following theorem was presented which gives conditions under which the system in general form will converge.

THEOREM 2.2. (General Index One WR Convergence Theorem[4]) *Consider a WR algorithm whose iterated equations can be transformed into the following canonical form*

$$\dot{\mathbf{w}}^{k+1} = \hat{F}_1(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k), \qquad \mathbf{w}^{k+1}(0) = \mathbf{w}_0, \tag{28}$$

$$\mathbf{z}^{k+1} = \hat{F}_2(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k), \tag{29}$$

where $\mathbf{w}^k$, $\mathbf{w}^{k+1} \in \mathbb{R}^n$, $\mathbf{z}^k$, $\mathbf{z}^{k+1} \in \mathbb{R}^m$ *and* $\mathbf{w}_0 \in \mathbb{R}^n$. *Assume that there exist norms in* $\mathbb{R}^n \times \mathbb{R}^m$ *and* $\mathbb{R}^n$, $\lambda_1 \geq 0$, $\lambda_2 \geq 0$, *and* $\gamma \in [0, 1)$ *such that for any* $\mathbf{a}, \mathbf{b}, \mathbf{s}, \tilde{\mathbf{a}}, \tilde{\mathbf{b}}, \tilde{\mathbf{s}} \in \mathbb{R}^n$ *and* $\mathbf{v}, \tilde{\mathbf{v}} \in \mathbb{R}^m$

$$\left\| \begin{matrix} \hat{F}_1(\mathbf{a}, \mathbf{b}, \mathbf{s}, \mathbf{v}) - \hat{F}_1\left(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}, \tilde{\mathbf{s}}, \tilde{\mathbf{v}}\right) \\ \hat{F}_2(\mathbf{a}, \mathbf{b}, \mathbf{s}, \mathbf{v}) - \hat{F}_2\left(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}, \tilde{\mathbf{s}}, \tilde{\mathbf{v}}\right) \end{matrix} \right\| \leq \lambda_1 \|\mathbf{a} - \tilde{\mathbf{a}}\| + \lambda_2 \left\|\mathbf{b} - \tilde{\mathbf{b}}\right\| + \gamma \left\| \begin{matrix} \mathbf{s} - \tilde{\mathbf{s}} \\ \mathbf{v} - \tilde{\mathbf{v}} \end{matrix} \right\|.$$

Then for any initial guesses $(\mathbf{w}^0(t), \mathbf{z}^0(t);\ t \in [0, T])$, *the sequence* $\{(\dot{\mathbf{w}}^{k+1}(t), \mathbf{w}^{k+1}(t), \mathbf{z}^{k+1}(t);\ t \in [0, T])\}_{k=1}^{\infty}$ *generated by the WR algorithm converges uniformly to* $(\dot{\hat{\mathbf{w}}}(t), \hat{\mathbf{w}}(t), \hat{\mathbf{z}}(t);\ t \in [0, T])$ *which satisfies*

$$\dot{\hat{\mathbf{w}}} = \hat{F}_1(\hat{\mathbf{w}}, \hat{\mathbf{w}}, \dot{\hat{\mathbf{w}}}, \hat{\mathbf{z}}), \qquad \hat{\mathbf{w}}(0) = \mathbf{w}_0, \tag{30}$$

$$\hat{\mathbf{z}} = \hat{F}_2(\hat{\mathbf{w}}, \hat{\mathbf{w}}, \dot{\hat{\mathbf{w}}}, \hat{\mathbf{z}}). \tag{31}$$

This theorem states that if the canonical form of the WR algorithm satisfies the conditions that the functions $(\hat{F}_1, \hat{F}_2)$ are globally contractive with respect to $(\dot{\mathbf{w}}, \mathbf{z})$ in some uniform norm, then the WR algorithm will converge. For most functions, the Lipschitz conditions are mild compared to the contractive conditions; thus, the constraining condition for convergence will usually be the contractivity requirement.

[3]This notion of *canonical form* is consistent with the definition in [19]. A comparison of this definition and the definition of *canonical form* found in [7] is given in Appendix A.

[4]The proof of this theorem may be found in [19] and is therefore omitted.

2.3. Convergence of the WR Algorithm for Index Two Systems

Several problems in engineering and physics result in problems which have an index of two. Applying the WR algorithm to index two systems yields the canonical form

$$\dot{\mathbf{w}}^{k+1} = f(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k, \dot{\mathbf{z}}^k), \tag{32}$$

$$\mathbf{z}^{k+1} = g(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k, \dot{\mathbf{z}}^k), \tag{33}$$

where $\mathbf{w} \in \mathbb{R}^{\hat{n}}$ is possibly (usually) different from $\mathbf{x}(t) \in \mathbb{R}^n$ with $\hat{n} \leq n$, $\mathbf{z}(t) \in \mathbb{R}^{\hat{m}}$ is possibly (usually) different from $\mathbf{y}(t) \in \mathbb{R}^m$ with $\hat{m} \geq m$, and f, g are continuous mapping functions of appropriate dimensions which are possibly (usually) different from F and G. It is shown in Appendix A that index two systems give rise to this canonical form. Note that, although similar, this canonical from is different from the canonical form presented in Theorem 2.2, and thus, Theorem 2.2 cannot be used to determine the convergence of a system which has this canonical form. In the remainder of this section, sufficient conditions to guarantee convergence of a WR algorithm are derived. The conditions are stated for the canonical form of the WR algorithm, but the WR is not necessarily implemented in its canonical form, i.e., it is not required to find f and g explicitly.

THEOREM 2.3. (Index Two Canonical WR Convergence Theorem) *Consider a WR algorithm whose iterated equations can be transformed into the following canonical form*

$$\dot{\mathbf{w}}^{k+1} = f(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k, \dot{\mathbf{z}}^k), \qquad \mathbf{w}^{k+1}(0) = \mathbf{w}_0, \tag{34}$$

$$\mathbf{z}^{k+1} = g(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k, \dot{\mathbf{z}}^k), \tag{35}$$

where $\mathbf{w}^k$, $\mathbf{w}^{k+1} \in \mathbb{R}^n$, $\mathbf{z}^k$, $\mathbf{z}^{k+1} \in \mathbb{R}^m$ *and* $\mathbf{w}_0 \in \mathbb{R}^n$. *Assume that*

1. *the canonical* differential *variables can be expressed in the form*

$$\dot{\mathbf{w}}^{k+1} = \tilde{F}(\mathbf{w}^{k+1}, \mathbf{w}^k, \mathbf{w}^{k-1}, \dot{\mathbf{w}}^k, \dot{\mathbf{w}}^{k-1});$$

2. *there exist norms in* $\mathbb{R}^n \times \mathbb{R}^n$, $\lambda_1 \geq 0$, $\lambda_2 \geq 0$, $\lambda_3 \geq 0$, *and* $\gamma_1 \in [0,1)$, $\gamma_2 \in [0,1)$ *where* $\gamma_1 + \gamma_2 < 1$, *such that for any* $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, $\mathbf{d}$, $\mathbf{e}$, $\tilde{\mathbf{a}}$, $\tilde{\mathbf{b}}$, $\tilde{\mathbf{c}}$, $\tilde{\mathbf{d}}$, $\tilde{\mathbf{e}} \in \mathbb{R}^n$

$$\left\| \tilde{F}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}) - \tilde{F}(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}, \tilde{\mathbf{c}}, \tilde{\mathbf{d}}, \tilde{\mathbf{e}}) \right\| \leq \lambda_1 \|\mathbf{a} - \tilde{\mathbf{a}}\| + \lambda_2 \left\| \mathbf{b} - \tilde{\mathbf{b}} \right\| + \lambda_3 \|\mathbf{c} - \tilde{\mathbf{c}}\| + \gamma_1 \left\| \mathbf{d} - \tilde{\mathbf{d}} \right\| + \gamma_2 \|\mathbf{e} - \tilde{\mathbf{e}}\|.$$

Then for any initial guesses $(\mathbf{w}^0(t), \mathbf{z}^0(t);\ t \in [0,T])$, *the sequence* $\{(\dot{\mathbf{w}}^{k+1}(t), \mathbf{w}^{k+1}(t), \mathbf{z}^{k+1}(t);\ t \in [0,T])\}_{k=1}^{\infty}$ *generated by the WR algorithm converges uniformly to* $(\dot{\hat{\mathbf{w}}}(t), \hat{\mathbf{w}}(t), \hat{\mathbf{z}}(t);\ t \in [0,T])$ *which satisfies*

$$\dot{\hat{\mathbf{w}}} = f(\hat{\mathbf{w}}, \hat{\mathbf{w}}, \dot{\hat{\mathbf{w}}}, \hat{\mathbf{z}}, \dot{\hat{\mathbf{z}}}), \qquad \hat{\mathbf{w}}(0) = \mathbf{w}_0, \tag{36}$$

$$\hat{\mathbf{z}} = g(\hat{\mathbf{w}}, \hat{\mathbf{w}}, \dot{\hat{\mathbf{w}}}, \hat{\mathbf{z}}, \dot{\hat{\mathbf{z}}}). \tag{37}$$

This theorem establishes sufficient conditions under which the WR algorithm will converge to the correct solution. Assumption 1 of Theorem 2.3 ensures that the system dynamic behavior is governed strictly by the independent time varying variables (i.e., the state variables) and the application of the WR simulation method does not introduce a dependence of the state variables on the initial waveform guess of an algebraic variable. This may be interpreted as a strong statement of contractivity of the algebraic variables with respect to the differential variables. Assumption 2 establishes the criteria under which the differential canonical variables will converge given Assumption 1. The significance of this theorem will be further discussed in Section 3.

Two illustrative examples are given in the next section which explicitly highlight the differences between this theorem and Theorem 2.2 for general form index one systems.

2.4. Illustrative Examples

Consider the following linear time invariant (LTI) index two system

$$\begin{aligned}
\dot{y}_1 &= -4y_1 + 2y_2 - y_3 + y_4 + 0.5y_5,\\
\dot{y}_2 &= y_1 - y_2 + y_3 - 0.5y_5,\\
0 &= -y_1 + y_2 + y_4,\\
\dot{y}_4 &= 1.25y_1 + 2.25y_2 - 4y_4 + y_5,\\
0 &= -0.5y_1 - 0.5y_2 + y_4,
\end{aligned}$$

which is partitioned as (y_1, y_2, y_3) (y_4, y_5) for the WR method yielding

$$\begin{aligned}
\dot{y}_1^{k+1} &= -4y_1^{k+1} + 2y_2^{k+1} - y_3^{k+1} + y_4^k + 0.5y_5^k,\\
\dot{y}_2^{k+1} &= y_1^{k+1} - y_2^{k+1} + y_3^{k+1} - 0.5y_5^k,\\
0 &= -y_1^{k+1} + y_2^{k+1} + y_4^k,\\
\dot{y}_4^{k+1} &= 1.25y_1^k + 2.25y_2^k - 4y_4^{k+1} + y_5^{k+1},\\
0 &= -0.5y_1^k - 0.5y_2^k + y_4^{k+1}.
\end{aligned}$$

This system is solvable, therefore there exists a canonical form

$$\dot{w}^{k+1} = -3w^{k+1} + z_3^k, \tag{38}$$
$$z_1^{k+1} = -z_3^k, \tag{39}$$
$$z_2^{k+1} = z_3^k + z_4^k - \dot{z}_3^k, \tag{40}$$
$$z_3^{k+1} = 0.5w^k, \tag{41}$$
$$z_4^{k+1} = 0.25w^k - 0.5z_1^k + 0.5\dot{w}^k, \tag{42}$$

where

$$\begin{aligned}
w &\triangleq y_1 + y_2,\\
z_1 &\triangleq -y_1 + y_2,\\
z_2 &\triangleq 5y_1 - 7y_2 + 2y_3,\\
z_3 &\triangleq y_4,\\
z_4 &\triangleq y_5.
\end{aligned}$$

The convergence of this system cannot be determined by Theorem 2.2 due to the $\dot{z}_3^k$ term in equation (40) which violates the statement of the theorem. However, equation (38) can be reduced to

$$\dot{w}^{k+1} = -3w^{k+1} + 0.5w^{k-1}, \tag{43}$$

which will guarantee convergence to the correct solution by Theorem 2.3.

A second illustrative example involves the following linear index two system:

$$\dot{y}_1 = -3y_1 + y_2 + y_3 - y_4, \tag{44}$$
$$\dot{y}_2 = y_1 - 2y_2, \tag{45}$$
$$\dot{y}_3 = y_2 - 4y_3 + y_5, \tag{46}$$
$$0 = y_1 + y_2 - 3y_3, \tag{47}$$
$$0 = 2y_4 + y_5. \tag{48}$$

Grouping equations ((44), (45), (48)) and ((46), (47)) and the variables (y_1, y_2, y_4) and (y_3, y_5) for the WR method, equations (44)–(48) become

$$\dot{y}_1^{k+1} = -3y_1^{k+1} + y_2^{k+1} + y_3^k - y_4^{k+1}, \tag{49}$$

$$\dot{y}_2^{k+1} = y_1^{k+1} - 2y_2^{k+1}, \tag{50}$$

$$0 = 2y_4^{k+1} + y_5^{k+1}, \tag{51}$$

$$\dot{y}_3^{k+1} = y_2^k - 4y_3^{k+1} + y_5^{k+1}, \tag{52}$$

$$0 = y_1^k + y_2^k - 3y_3^{k+1}. \tag{53}$$

Note that the decomposed system now contains one index-1 subsystem and one index-2 subsystem. This system is solvable, therefore it can be transformed into the following canonical form

$$\dot{w}_1^{k+1} = -3w_1^{k+1} + w_2^{k+1} + z_1^k + \frac{1}{2}z_3^k, \tag{54}$$

$$\dot{w}_2^{k+1} = w_1^{k+1} - 2w_2^{k+1}, \tag{55}$$

$$z_1^{k+1} = \frac{1}{3}w_1^k + \frac{1}{3}w_2^k, \tag{56}$$

$$z_2^{k+1} = -\frac{1}{2}z_3^k, \tag{57}$$

$$z_3^{k+1} = \frac{4}{3}w_1^k + \frac{1}{3}w_2^k + \frac{1}{3}\dot{w}_1^k + \frac{1}{3}\dot{w}_2^k, \tag{58}$$

where

$$\begin{aligned} w_1 &\triangleq y_1, \\ w_2 &\triangleq y_2, \\ z_1 &\triangleq y_3, \\ z_2 &\triangleq y_4, \\ z_3 &\triangleq y_5. \end{aligned}$$

Since no $\dot{z}^k$ terms appear in the canonical system, the Theorem 2.3 sufficiency conditions may be implemented. Unfortunately however, the convergence of the system cannot be determined since the contractivity requirement

$$\left\| \begin{bmatrix} 0 & 0 & 1 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{2} \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \end{bmatrix} \right\| < 1$$

is not satisfied. It is noticed however, that the canonical differential variables may be rewritten as

$$\begin{aligned} \dot{w}_1^{k+1} &= -3w_1^{k+1} + w_2^{k+1} + w_1^{k-1} + \frac{1}{2}w_2^{k-1} + \frac{1}{6}\dot{w}_1^{k-1} + \frac{1}{6}\dot{w}_2^{k-1}, \\ \dot{w}_2^{k+1} &= w_1^{k+1} - 2w_2^{k+1}, \end{aligned}$$

which satisfy the contractivity requirements of Theorem 2.3 that

$$\left\| \begin{bmatrix} \frac{1}{6} & \frac{1}{6} \\ 0 & 0 \end{bmatrix} \right\| < 1.$$

Therefore, the sufficiency requirements for convergence are satisfied, and the WR algorithm of equations (49)–(53) will converge.

At this point, one final note to this section on the index two WR method is pointed out. Whether or not the WR iterates will converge depends very heavily on the manner in which the original problem is partitioned. For a particular partitioning the system may not be solvable, or may result in a nonconvergent algorithm. As an example of the latter problem, consider again the previous linear index two system of equations (44)–(48). If this had been partitioned differently as ((44), (45), (47)) and ((46), (48)), the system is solvable and yields the canonical form

$$\begin{aligned}
\dot{w}_1^{k+1} &= -3w_1^{k+1} + 3w_2^{k+1}, \\
z_1^{k+1} &= 3w_2^k, \\
z_2^{k+1} &= -w_2^k + 3\dot{w}_2^k, \\
\dot{w}_2^{k+1} &= -w_1^k - 4w_2^{k+1} + 4z_1^k + 2z_2^k, \\
z_3^{k+1} &= -2w_1^k + 4z_1^k + 2z_2^k,
\end{aligned}$$

where

$$\begin{aligned}
w_1 &\triangleq y_2, \\
w_2 &\triangleq -y_3, \\
z_1 &\triangleq y_1 + y_2, \\
z_2 &\triangleq -2y_1 - y_2 - y_4, \\
z_3 &\triangleq y_5.
\end{aligned}$$

Once again, the sufficiency criterion of Theorem 2.2

$$\left\| \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 4 & 2 & 0 \\ 0 & 0 & 4 & 2 & 0 \end{bmatrix} \right\| < 1$$

is not satisfied. Similarly, even though a reduced canonical form for the differential variables can be found

$$\begin{aligned}
\dot{w}_1^{k+1} &= -3w_1^{k+1} + 3w_2^k, \\
\dot{w}_2^{k+1} &= -w_1^k - 4w_2^{k+1} + 12w_2^{k-1} + 6\dot{w}_2^{k-1},
\end{aligned}$$

the sufficiency criterion for convergence of the iterates by Theorem 2.3 that

$$\left\| \begin{bmatrix} 0 & 0 \\ 0 & 6 \end{bmatrix} \right\| < 1$$

is not satisfied as well. This system does indeed fail to converge for this particular partitioning. From this example, it is concluded that the partitioning of the original system is important for convergence.

Theorem 2.3 may be generalized to other index systems as well. For higher index systems, the convergence theorem follows that of the index two case, except the canonical differential functions must be m-point contractive with respect to the time derivative iterates, where m is the index of the system.[5]

[5]Two-point contractivity is defined in Appendix B.

2.5. Nonstationary WR Algorithms

To this point the WR algorithms presented have been *stationary* algorithms in the sense that the iteration process is performed with the same set of equations. In practice, however, the equations change slightly from iteration to iteration. These changes are a result of the finite error introduced when the nonlinear equations are not solved exactly. These errors can be controlled, but not altogether eliminated. *Nonstationary* algorithms are algorithms in which the equations describing the system at each iteration differ from one iteration to the next. To be considered useful, the nonstationary algorithms must approach the stationary function as $k \to \infty$. In the remainder of the section, it will be shown that the nonstationary WR algorithms converge as a direct consequence of the contraction mapping property of the original WR algorithm. This result will be developed for the index two case from which the result can be generalized to other index cases.

THEOREM 2.4. *Let F be a mapping in a Banach space Y which describes the reduced canonical system of Theorem 2.3. Let $F^k : Y \to Y$ be the nonstationary mapping approximating F at the k^{th} iteration. Assume that F is a contraction mapping in the two-point sense with contraction factors α_1 and α_2 $(\alpha_1 + \alpha_2 < 1)$ and $\|F^k(y) - F(y)\| \leq \delta^k$ for all $y \in Y$ and define $y^{k+1} = F(y^k)$ and $\tilde{y}^{k+1} = F^k(\tilde{y}^k)$. If there exists a $y^* \in Y$ such that $F(y^*) = y^*$, then for any $\epsilon > 0$ there exists a $\delta(\epsilon) < 1$ such that if $\delta^k < \delta$ for all k then $\lim_{k\to\infty} \|\tilde{y}^k - \tilde{y}^{k-1}\| < \epsilon$ and $\lim_{k\to\infty} \|\tilde{y}^k - y^*\| < \delta/(1 - (\alpha_1 + \alpha_2))$.*

This theorem essentially states that any fairly accurate approximation to the function can be used to solve the WR iterates as long as the subsequent error is driven to zero as the iterations progress. It is difficult to ascertain beforehand how precise "fairly accurate" must be to meet this criteria since this depends on the contraction factors α_1 and α_2 which are difficult to estimate in many cases.

3. THE DISCRETIZED WAVEFORM RELAXATION METHOD FOR DIFFERENTIAL/ALGEBRAIC SYSTEMS

A major advantage of the WR algorithm is that the differential/algebraic equations are solved in a decomposed manner. This implies that if discretization methods are used to solve the independent sets of equations, the time steps for each subsystem can be selected relatively independently. This leads naturally to three questions: Does the waveform relaxation process still converge? If it does converge, does the resulting multirate integration method possess the stability properties of the integration method used for the decomposed systems? Can the various time steps be chosen independently?

The discretized WR algorithm is a nonstationary method, thus the theorem presented in Section 4.3 may be applied to insure that the WR iterates will converge to the solution of a given system of DAEs when the *global* discretization error is driven to zero as the number of iterations increases. The time steps for numerical integration methods are usually chosen to guarantee that local truncation error estimates are kept below some preset threshold. However, in certain instances, it may be possible to satisfy the local truncation error criterion, but the discretized WR algorithm may still fail to converge. This situation implies that the discretization process is more complex in the WR algorithm than for classical numerical integration methods. The discretized WR algorithm for ODEs is discussed in [17,23].

3.1. The Uniform Time Step Case

As an example of a situation where the WR convergence depends on discretization, consider WR applied to the following linear index two test system with partitioning $\{x_1, y\}, \{x_2\}$

$$\dot{x}_1 = -5x_1 + y + 0.1x_2, \tag{59}$$

$$0 = x_1 + \lambda_1 x_2, \tag{60}$$

$$\dot{x}_2 = x_1 - \lambda_2 x_2. \tag{61}$$

This system is solvable and stable for all $\lambda_1, \lambda_2 \geq 0$. Therefore, if each equation in the system is discretized identically (this is known as a *uniform* time step), the discretized equations under backward-Euler become

$$(1 + 5h)\, x_{1,n+1}^{k+1} - h\, y_{n+1}^{k+1} = 0.1\, h\, x_{2,n+1}^{k} + x_{1,n}^{k+1}, \tag{62}$$

$$x_{1,n+1}^{k+1} = -\lambda_1\, x_{2,n+1}^{k}, \tag{63}$$

$$(1 + \lambda_2\, h)\, x_{2,n+1}^{k+1} = h\, x_{1,n+1}^{k} + x_{2,n}^{k+1}, \tag{64}$$

where $h = t_{n+1} - t_n$.

The waveforms for the Gauss-Jacobi WR algorithm exhibit a strange behavior when this algorithm is applied to DAE systems of index greater than or equal to two. The iterates for all variables do not change at every iteration as they might in the WR applied to ODEs, but for an index two case, they only change every other iteration. The first few iterations of x_1 and x_2 for $h = 0.01$ of the previous example are given in Table 1, with the chosen initial condition. From Table 1, note that x_1 is updated in iterations 2 and 4, whereas x_2 is updated in iterations 1 and 3.

Table 1. Successive iterates for the Linear Index Two Test System.

	$k=1$		$k=2$		$k=3$		$k=4$	
t	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2
0.00	−1.0000	0.1000	−1.0000	0.1000	−1.0000	0.1000	−1.0000	0.1000
0.01	−1.0000	0.0895	−0.8955	0.0895	−0.8955	0.0905	−0.9059	0.0905
0.02	−1.0000	0.0791	−0.7915	0.0791	−0.7915	0.0822	−0.8226	0.0822
0.03	−1.0000	0.0688	−0.6881	0.0688	−0.6881	0.0750	−0.7500	0.0750
0.04	−1.0000	0.0585	−0.5851	0.0585	−0.5851	0.0688	−0.6881	0.0688
0.05	−1.0000	0.0482	−0.4827	0.0482	−0.4827	0.0636	−0.6366	0.0636

An examination of the discretized equations for the example clarifies why this phenomenon occurs. After discretization by backward-Euler, the discrete variables become

$$x_{1,n+1}^{k+1} = -\lambda_1\, x_{2,n+1}^{k}, \tag{65}$$

$$y_{n+1}^{k+1} = -\left(0.1 + \lambda_1 \left(\frac{1+5h}{h}\right)\right) x_{2,n+1}^{k} - \frac{1}{h}\, x_{1,n}^{k+1}, \tag{66}$$

$$x_{2,n+1}^{k+1} = \frac{h}{1+\lambda_2 h}\, x_{1,n+1}^{k} + \frac{1}{1+\lambda_2 h}\, x_{2,n}^{k+1}. \tag{67}$$

Thus $x_{1,n+1}^{k+1}$ can be equivalently expressed as

$$x_{1,n+1}^{k+1} = -\lambda_1 \left(\frac{h}{1+\lambda_2 h}\, x_{1,n+1}^{k-1} + \frac{1}{1+\lambda_2 h}\, x_{2,n}^{k}\right),$$

and similarly

$$x_{2,n+1}^{k+1} = \frac{h}{1+\lambda_2 h}\left(-\lambda_1\, x_{2,n+1}^{k-1}\right) + \frac{1}{1+\lambda_2 h}\, x_{2,n}^{k+1}.$$

Note the dependence of the $(k+1)^{\text{st}}$ iterate on the $(k-1)^{\text{st}}$ iterate. Thus, each of these variables is only updated every other iteration, and the "middle" iteration is redundant. This dependency is not unexpected in light of Convergence Theorem 2.3 which states that an index two system may give rise to this type of "nested" iterations. This redundancy can be eliminated directly by computing only every other iteration without loss of information.

This phenomenon also occurs in Gauss-Seidel based iterations, although it is not as apparent as with the Gauss-Jacobi iterations. During G-S iterations, the $(k+1)^{st}$ iterate will, in general, depend on both the k^{th} and the $(k-1)^{st}$ iterates. Thus the "nesting" effect is still occurring, leading to possibly slower convergence of the iterates, but it is more difficult to detect than with the G-J iterations.

The algebraic relaxation of this system will converge if the eigenvalues of the relaxation matrix lie within the unit circle in the complex plane. For the algebraic relaxation this leads to the following allowable values of h

$$h < \frac{1}{\lambda_1 - \lambda_2}, \qquad \lambda_1 > \lambda_2 \geq 0.$$

For example, if $\lambda_1 = 10$ and $\lambda_2 = 0.5$, then

$$h < \frac{1}{10 - 0.5} \approx 0.105.$$

To see how this h might be related to the allowable time steps for the WR algorithm, $h = 0.1$ sec is used in the above example. Using an initial guess $x^0(t) = x_0;\ t \in [0, T]$, the WR iterates converge to the solution of the discretized equations. However, when the time step h is increased slightly to 0.11, the iterates diverge for any choice of initial waveform. This implies that there is a close relationship between the time step bound for the algebraic relaxation and the time step bound for the WR algorithm.

The WR iterates given above will converge to the discretized solution if

$$\frac{\lambda_1 h}{1 + \lambda_2 h} < 1,$$

or equivalently

$$h < \frac{1}{\lambda_1 - \lambda_2}.$$

This is identical to the time step bounds given by the algebraic relaxation method. The upper bounds on the time steps for which the uniform time step discretized WR algorithm will converge are very similar to the constraints on the time steps for which the algebraic relaxation algorithm applied to a discretized numerical integration scheme will converge. In fact, in the linear case they are closely related as was shown in the previous example. A comparison theorem between the WR and the algebraic relaxation method is presented below.

THEOREM 3.1. *Let a k-step backwards difference formula be applied to the linear DAE system of the form*

$$\mathbf{E}\,\dot{\mathbf{y}}(t) = \mathbf{A}\,\mathbf{y}(t), \qquad \mathbf{y}(0) = \mathbf{y}_0,$$

where $\mathbf{E}$, $\mathbf{A} \in \mathbb{R}^{m \times m}$, *and* $\mathbf{E}$ *has the form*

$$\mathbf{E} = \mathrm{diag}\{\mathbf{E}_i\}, \qquad i = 1, \ldots, r,$$

where

$$\mathbf{E}_i = \begin{bmatrix} \mathbf{I}_{n_i} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \sum_i^r n_i = n; \qquad n \leq m$$

and $\mathbf{y}(t) \in \mathbb{R}^m$. *Assume that the Gauss-Jacobi (or Gauss-Seidel) algebraic relaxation algorithm is used to solve the linear algebraic equations generated by the BDF. Given a sequence of time steps* $\{h_m\}$, *where each* h_m *is chosen such that the discretized DAE is* h_m *solvable, the Gauss-Jacobi (Gauss-Seidel) relaxation algorithm will converge at every step, for any initial guess, if and only if the WR algorithm, discretized with the same sequence of time steps and with the same BDF, converges for any initial guess.*

This theorem may be generalized to the nonlinear WR algorithm if it is assumed that the initial guess may by chosen arbitrarily close to the exact solution. This requirement is necessary because of the constraints imposed by the Newton's method on the initial guess.

3.2. The Multirate WR Convergence Theorem

A major advantage of the WR algorithm for both ODEs and DAEs is that the differential/algebraic equations are solved in a decomposed fashion. This implies that if discretization methods are used to solve the independent subsystems, the time steps used by each subsystem may be selected fairly independently. When different time steps are chosen by the individual subsystems, the computation of values at a time step of one subsystem may require a value of another subsystem which was not explicitly computed, due to differing time steps. This value must then be interpolated. If this interpolation is not performed carefully, the convergence of the algorithm may be destroyed. To insure convergence of the WR algorithm, the error introduced by the interpolation must be driven to zero with the WR iterations in accordance with the theorem presented in Section 4.3 for nonstationary methods.

The most common type of interpolation is linear interpolation, where the unknown value is taken to be on a line connecting the immediate greater and lesser values. For example, if $\hat{x}(\tau_{n-1})$ and $\hat{x}(\tau_n)$ are known, then $\hat{x}(\hat{\tau})$ is approximated by

$$\hat{x}(\hat{\tau}) = \left(\frac{\hat{\tau} - \tau_{n-1}}{\tau_n - \tau_{n-1}}\right)(\hat{x}(\tau_n) - \hat{x}(\tau_{n-1})) + \hat{x}(\tau_{n-1}),$$

where $\tau_{n-1} \leq \hat{\tau} \leq \tau_n$. If linear interpolation is used to approximate the unknown values, the multirate discretized WR algorithm will converge. This statement is formalized in the following theorem.

THEOREM 3.2. *If linear interpolation is used to approximate the unknown variable values for nonconcurrent time steps, then there exists a collection of time steps $h_{i0} > 0$, $i = \{1, \ldots, n\}$, such that if $0 < h_i \leq h_{i0}$ for all i, then the multirate fixed-time step discretized WR algorithm converges with respect to the interpolated sequences.*[6]

This theorem implies that the discretized WR algorithms will converge only if the "underlying" discretized equations will converge. When a DAE system of index m is divided into subsystems, not all of the subsystems may have index m, some may have index less than m. This implies that different restrictions may apply in discretizing the various subsystems. A subsystem having an index of three must be discretized by a constant step size BDF of order ≤ 6 [13], whereas, a subsystem of index two or less may be discretized by a variable-step variable-order BDF [10]. This is an additional advantage of the WR method. Those subsystems which are of low index may be integrated with traditional numerical methods with little or no additional precautionary measures, while those of higher index may be integrated using methods tailored for high index systems.

4. CONCLUSIONS

Many systems in robotics and control applications are modeled with DAE systems having an index greater than two. Computer simulation of these systems has been hampered by numerical integration methods which perform poorly and must be explicitly tailored to the system. The WR algorithm presents a means by which these systems may be more efficiently simulated by breaking them into weakly coupled subsystems, many of which will no longer retain the limiting high-index properties. This analysis also explains why VLSI circuits with floating capacitors encounter convergence difficulties. These difficulties are due to the "nesting" behavior of the iterations as a result of the increased index. Once these problems are identified, they may be eliminated. This paper presents an extension of the waveform relaxation algorithm to systems

[6]The proof of this theorem parallels a similar proof in [17] for ODEs. For an index two DAE system, the multirate discretized WR differential canonical system is a two-point contraction in an exponential norm. This result is used to prove convergence. This result may be generalized to other index systems provided the contraction assumptions are generalized as well.

of differential/algebraic equations. Although this type of application has been explored earlier in relation to VLSI circuits, the algorithm has not been generalized to include the vast array of DAE system structures. This paper establishes the solvability and convergence requirements of the waveform relaxation algorithm for higher-index systems.

APPENDIX A
DISCUSSION OF THE CANONICAL FORM FOR WAVEFORM RELAXATION APPLICATIONS

This appendix is included to avoid any confusion which might arise due to the difference in the usage of the term "canonical form" of a DAE as used in [7] versus the same terminology defined in [19]. For simplicity, this appendix will discuss canonical forms for linear time invariant DAE systems, but the results may be extended to the nonlinear DAE system.

Consider a *solvable* LTI DAE system

$$\mathbf{E}\,\dot{\mathbf{y}}(t) = \mathbf{A}\,\mathbf{y}(t) + \mathbf{g}(t), \qquad \mathbf{y}(t) \in \mathbb{R}^n, \tag{68}$$

where $\mathbf{E}$ is a noninvertible square matrix. As stated in [5,7], there exist nonsingular matrices $\mathbf{P}$ and $\mathbf{Q}$ such that

$$\mathbf{P}\,(\mathbf{A} - \lambda\mathbf{E})\,\mathbf{Q} = \begin{bmatrix} \mathbf{E}_1 + \lambda\mathbf{I}_{n_1} & 0 \\ 0 & \mathbf{I}_{n_2} + \lambda\mathbf{E}_2 \end{bmatrix}, \tag{69}$$

where $\mathbf{E}_1$ and $\mathbf{E}_2$ are $n_1 \times n_1$ and $n_2 \times n_2$ respectively, and $n = n_1 + n_2$ and $\mathbf{E}_2$ has the property that either there is an integer m such that $\mathbf{E}^m \equiv 0; \mathbf{E}^{m-1} \neq 0$ or $\mathbf{E} \equiv 0$. The integer m is defined to be the *index* of the system [13]. The matrices $\mathbf{P}$ and $\mathbf{Q}$ may be applied to (68),

$$\mathbf{P}\,\mathbf{E}\,\mathbf{Q}\,\mathbf{Q}^{-1}\,\dot{\mathbf{y}}(t) = \mathbf{P}\,\mathbf{A}\,\mathbf{Q}\,\mathbf{Q}^{-1}\,\mathbf{y}(t) + \mathbf{P}\,\mathbf{g}(t), \tag{70}$$

which decouples the system into a *canonical form* as defined in [5,7],

$$\dot{\mathbf{w}}(t) = \mathbf{E}_1\mathbf{w}(t) + \mathbf{f}_1(t), \tag{71}$$

$$\mathbf{E}_2\dot{\mathbf{z}}(t) = \mathbf{z}(t) + \mathbf{f}_2(t), \tag{72}$$

where $\mathbf{E}_1$ and $\mathbf{E}_2$ are from (69), $[\mathbf{w}(t)\ \mathbf{z}(t)]^\top = \mathbf{Q}^{-1}\,\mathbf{y}(t)$, and $[\mathbf{f}_1(t)\ \mathbf{f}_2(t)]^\top = \mathbf{P}\,\mathbf{g}(t)$. Thus (72) may be solved

$$\begin{aligned}\mathbf{z}(t) &= -\sum_{i=0}^{m-1} \mathbf{E}_2^i\,\mathbf{f}_2^{(i)}(t) \\ &= -\mathbf{f}_2(t) - \mathbf{E}_2\,\mathbf{f}_2^{(1)}(t) - \mathbf{E}_2^2\,\mathbf{f}_2^{(2)}(t) - \cdots - \mathbf{E}_2^{m-1}\,\mathbf{f}_2^{(m-1)}(t),\end{aligned}$$

where the superscript (i) denotes $\frac{d^{(i)}(\cdot)}{dt^{(i)}}$, whereas the superscript i is simply "to the power i." Note that in an index two system, only the first two terms appear since $\mathbf{E}_2^2 \equiv 0$. Thus an index two LTI system of the form (71), (72), may be written

$$\dot{\mathbf{w}}(t) = \mathbf{E}_1\,\mathbf{w}(t) + \mathbf{f}_1(t), \tag{73}$$

$$\mathbf{z}(t) = -\mathbf{f}_2(t) - \mathbf{E}_2\,\dot{\mathbf{f}}_2(t). \tag{74}$$

Now, consider an LTI system to which WR has been applied,

$$\mathbf{E}\,\dot{\mathbf{y}}^{k+1}(t) = \mathbf{M}\,\mathbf{y}^{k+1}(t) + \mathbf{N}\,\mathbf{y}^k(t) + \hat{\mathbf{g}}(t), \tag{75}$$

where $\mathbf{A} = \mathbf{M} + \mathbf{N}$. This may be rewritten

$$\mathbf{E}\,\dot{\mathbf{y}}^{k+1}(t) = \mathbf{M}\,\mathbf{y}^{k+1}(t) + \mathbf{g}(t). \tag{76}$$

Note that $\mathbf{g}(t)$ in (68) is now $-\mathbf{N}\,\mathbf{y}^k(t) + \hat{\mathbf{g}}(t)$, thus both $\mathbf{f}_1(t)$ and $\mathbf{f}_2(t)$ will be functions of $\hat{\mathbf{g}}(t)$ and $\mathbf{N}\,\mathbf{y}^k(t)$. Equation (75) may now be written as in (73) and (74) with

$$\begin{aligned}\begin{bmatrix} \mathbf{F}_1(t) \\ \mathbf{f}_2(t) \end{bmatrix} &= \mathbf{P}\,\mathbf{N}\,\mathbf{Q}\begin{bmatrix} \mathbf{w}^k \\ \mathbf{z}^k \end{bmatrix} + \mathbf{P}\,\hat{\mathbf{g}}(t) \\ &= \hat{\mathbf{N}}\begin{bmatrix} \mathbf{w}^k \\ \mathbf{z}^k \end{bmatrix} + \mathbf{P}\,\hat{\mathbf{g}}(t) \\ &= \begin{bmatrix} \hat{\mathbf{N}}_{ww}\,\mathbf{w}^k + \hat{\mathbf{N}}_{wz}\,\mathbf{z}^k + \hat{\mathbf{f}}_1 \\ \hat{\mathbf{N}}_{zw}\,\mathbf{w}^k + \hat{\mathbf{N}}_{zz}\,\mathbf{z}^k + \hat{\mathbf{f}}_2 \end{bmatrix},\end{aligned}$$

where $\left[\hat{\mathbf{f}}_1(t)\ \hat{\mathbf{f}}_2(t)\right]^{\mathsf{T}} = \mathbf{P}\,\hat{\mathbf{g}}(t)$. This relationship leads to the following set of canonical equations

$$\dot{\mathbf{w}}^{k+1}(t) = \mathbf{E}_1\,\mathbf{w}^{k+1}(t) + \hat{\mathbf{f}}_1(t) + \hat{\mathbf{N}}_{ww}\,\mathbf{w}^k(t) + \hat{\mathbf{N}}_{wz}\,\mathbf{z}^k(t) \tag{77}$$

$$\begin{aligned}\mathbf{z}^{k+1}(t) = -\hat{\mathbf{f}}_2(t) - \hat{\mathbf{N}}_{zw}\,\mathbf{w}^k(t) - \hat{\mathbf{N}}_{zz}\,\mathbf{z}^k(t) - \mathbf{E}_2\dot{\hat{\mathbf{f}}}_2(t) \\ - \mathbf{E}_2\,\hat{\mathbf{N}}_{zz}\,\dot{\mathbf{w}}^k(t) - \mathbf{E}_2\,\hat{\mathbf{N}}_{zz}\,\dot{\mathbf{z}}^k(t).\end{aligned} \tag{78}$$

Thus the canonical form for WR applications can be generally stated

$$\dot{\mathbf{w}}^{k+1} = f(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k, \dot{\mathbf{z}}^k), \qquad \mathbf{w}^{k+1}(0) = \mathbf{w}_0, \tag{79}$$

$$\mathbf{z}^{k+1} = g(\mathbf{w}^{k+1}, \mathbf{w}^k, \dot{\mathbf{w}}^k, \mathbf{z}^k, \dot{\mathbf{z}}^k). \tag{80}$$

Note the appearance of both the $\dot{\mathbf{w}}^k$ and $\dot{\mathbf{z}}^k$ terms. The primary difference between Theorem 2.2 and Theorem 2.3 is the addition of the $\dot{\mathbf{z}}^k$ term, which may considerably impact the determination of convergence.

APPENDIX B
PROOF OF THEOREMS

DEFINITION APPENDIX B.1. *A mapping $f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ is* **two-point contractive** *on a set $D_0 \subset D$ if there are $\alpha_0 \in [0,1)$, $\alpha_1 \in [0,1)$, $\alpha_0 + \alpha_1 < 1$ such that*

$$\|f(x_0, x_1) - f(y_0, y_1)\| \le \alpha_0\,\|x_0 - y_0\| + \alpha_1\,\|x_1 - y_1\|, \tag{81}$$

for all x_0, x_1, y_0, $y_1 \in D_0$.

THEOREM APPENDIX B.1. (Two-point Contraction Mapping Theorem) *Suppose that $f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ is two-point contractive on a closed set $D_0 \subset D$ and that $f(D_0, D_0) \subset D_0$. Thus f has a unique fixed point x^* in D_0. Furthermore, for any initial guesses x^1, $x^0 \subset D_0$, x^1, x^0 linearly independent, the sequence $\{x^{k+1} = f(x^k, x^{k-1})\}$ converges uniformly to x^*.*

PROOF. Let x^1, x^0 be arbitrary points in D_0, and form the sequence $x^{k+1} = f(x^k, x^{k-1})$, $k = 1, 2, \ldots$. The function $f(D_0, D_0) \subset D_0$, $\{x^{k+1}\}$ is well-defined and lies in D_0, and let $\epsilon^{k+1} \triangleq \|x^{k+1} - x^*\|$. Thus $\epsilon^{k+1} \le \alpha_1\,\epsilon^k + \alpha_0\,\epsilon^{k-1}$, so that

$$\epsilon^{k+1} \le \left(\sum_{j=0}^{k/2} \alpha_1^{(k-2j)}\,\alpha_0^j\,C_j\right)\epsilon^1 + \alpha_0\left(\sum_{j=0}^{k/2} \alpha_1^{(k-2j-1)}\,\alpha_0^j\,C_j\right)\epsilon^0, \tag{82}$$

where

$$C_j = \sum_{i_j=0}^{k-2j}\ \sum_{i_{j-1}=0}^{i_j} \cdots \sum_{i_1=0}^{i_2} 1.$$

Thus, equation (82) may be rewritten as

$$\epsilon^{k+1} \leq \gamma_1(k)\,\epsilon^1 + \gamma_2(k)\,\epsilon^0. \tag{83}$$

Since $\gamma_1(k) \to 0$ as $k \to \infty$ and $\gamma_2(k) \to 0$ as $k \to \infty$, then as $k \to \infty$, $\epsilon^{k+1} \to 0$, thus $x^{k+1} \to x^*$ as $k \to \infty$. Therefore x^* is a fixed point of f. Now to prove that x^* is unique. Suppose x^* and y^* are two fixed points, then

$$\begin{aligned}\|x^* - y^*\| = \|f(x^*, x^*) - f(y^*, y^*)\| &\leq \alpha_1 \|x^* - y^*\| + \alpha_2 \|x^* - y^*\| \\ &\leq (\alpha_1 + \alpha_2) \|x^* - y^*\| < \|x^* - y^*\|.\end{aligned}$$

Hence a contradiction, therefore x^* is unique. ∎

This leads to the following lemma.

LEMMA APPENDIX B.1. *Let* $\mathbf{x}, \mathbf{y}, \mathbf{z} \in C([0,T], \mathbb{R}^n)$. *If there exists some norm on* $\mathbb{R}^n$ *such that*

$$\|\dot{\mathbf{x}}(t)\| \leq \gamma_1 \|\dot{\mathbf{y}}(t)\| + \gamma_2 \|\dot{\mathbf{z}}(t)\| + \ell_1 \|\mathbf{x}(t)\| + \ell_2 \|\mathbf{y}(t)\| + \ell_3 \|\mathbf{z}(t)\|, \tag{84}$$

for some positive numbers ℓ_1, ℓ_2, ℓ_3 *and* $\gamma_1 + \gamma_2 < 1$, *then there exists a norm* $\|\cdot\|_b$ *on* $C([0,t], \mathbb{R}^n)$ *such that*

$$\|\dot{\mathbf{x}}(t)\|_b \leq \alpha_1 \|\dot{\mathbf{y}}(t)\|_b + \alpha_2 \|\dot{\mathbf{z}}(t)\|_b + L_1 \|\mathbf{x}(0)\| + L_2 \|\mathbf{y}(0)\| + L_3 \|\mathbf{z}(0)\|, \tag{85}$$

for some positive α_1, α_2, L_1, L_2, L_3 *such that* $\alpha_1 + \alpha_2 < 1$.

PROOF. The proof of this lemma may be found in [21] and is omitted for brevity. This proof is close in development to a similar proof for ODEs presented in [17].

PROOF OF THEOREM 2.3. Given

$$\dot{\mathbf{w}}^{k+1} = \tilde{F}\left(\mathbf{w}^{k+1}, \mathbf{w}^k, \mathbf{w}^{k-1}, \dot{\mathbf{w}}^k, \dot{\mathbf{w}}^{k-1}\right). \tag{86}$$

Taking the difference between equation (85) at iteration $k+1$ and at iteration $j+1$ yields

$$\dot{\mathbf{w}}^{k+1} - \dot{\mathbf{w}}^{j+1} = f(\mathbf{w}^{k+1}, \mathbf{w}^k, \mathbf{w}^{k-1}, \dot{\mathbf{w}}^k, \dot{\mathbf{w}}^{k-1}) - f(\mathbf{w}^{j+1}, \mathbf{w}^j, \mathbf{w}^{j-1}, \dot{\mathbf{w}}^j, \dot{\mathbf{w}}^{j-1}). \tag{87}$$

Using the Lipschitz continuity of f with respect to $\mathbf{w}$, and the contractivity with respect to $\dot{\mathbf{w}}$, (87) leads to

$$\begin{aligned}\left\|\dot{\mathbf{w}}^{k+1} - \dot{\mathbf{w}}^{j+1}\right\| \leq \gamma_1 &\left\|\dot{\mathbf{w}}^k - \dot{\mathbf{w}}^j\right\| + \gamma_2 \left\|\dot{\mathbf{w}}^{k-1} - \dot{\mathbf{w}}^{j-1}\right\| \\ &+ \ell_1 \left\|\mathbf{w}^{k+1} - \mathbf{w}^{j+1}\right\| + \ell_2 \left\|\mathbf{w}^k - \mathbf{w}^j\right\| + \ell_3 \left\|\mathbf{w}^{k-1} - \mathbf{w}^{j-1}\right\|.\end{aligned} \tag{88}$$

Then from Lemma Appendix B.1, there exists a norm such that

$$\begin{aligned}\left\|\dot{\mathbf{w}}^{k+1} - \dot{\mathbf{w}}^{j+1}\right\|_b \leq \alpha_1 &\left\|\dot{\mathbf{w}}^k - \dot{\mathbf{w}}^j\right\|_b + \alpha_2 \left\|\dot{\mathbf{w}}^{k-1} - \dot{\mathbf{w}}^{j-1}\right\|_b + L_1 \left\|\mathbf{w}^{k+1}(0) - \mathbf{w}^{j+1}(0)\right\| \\ &+ L_2 \left\|\mathbf{w}^k(0) - \mathbf{w}^j(0)\right\| + L_3 \left\|\mathbf{w}^{k-1}(0) - \mathbf{w}^{j-1}(0)\right\|,\end{aligned} \tag{89}$$

thus, since $\mathbf{w}^{k+1}(0) = \mathbf{w}^{j+1}(0)$, $\mathbf{w}^k(0) = \mathbf{w}^j(0)$, and $\mathbf{w}^{k-1}(0) = \mathbf{w}^{j-1}(0)$,

$$\left\|\dot{\mathbf{w}}^{k+1} - \dot{\mathbf{w}}^{j+1}\right\|_b \leq \alpha_1 \left\|\dot{\mathbf{w}}^k - \dot{\mathbf{w}}^j\right\|_b + \alpha_2 \left\|\dot{\mathbf{w}}^{k-1} - \dot{\mathbf{w}}^{j-1}\right\|_b. \tag{90}$$

Thus for $\alpha_1 + \alpha_2 < 1$, this is contractive by Definition 4.1. Hence, by Theorem Appendix B.1, it has a unique fixed point satisfying

$$\dot{\mathbf{w}}^* = f(\mathbf{w}^*, \mathbf{w}^*, \mathbf{w}^*, \dot{\mathbf{w}}^*, \dot{\mathbf{w}}^*), \qquad \mathbf{w}^*(0) = \mathbf{w}_0.$$

Furthermore, for any given initial $\mathbf{w}^1(0)$, $\mathbf{w}^0(0)$ the sequence $\left\{\mathbf{w}^{k+1}(\cdot)\right\}_{k=2}^{\infty}$ generated by (86) converges uniformly to $\dot{\mathbf{w}}^*$. ∎

PROOF OF THEOREM 2.4. Taking the difference between the k^{th} and the $(k+1)^{\text{st}}$ iterations of the nonstationary algorithm yields

$$\begin{aligned}\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\| &= \left\|F^{k+1}\left(\tilde{y}^{k},\tilde{y}^{k-1}\right)-F^{k}\left(\tilde{y}^{k-1},\tilde{y}^{k-2}\right)\right\| \qquad (91)\\ &\le \left\|F^{k+1}\left(\tilde{y}^{k},\tilde{y}^{k-1}\right)-F\left(\tilde{y}^{k},\tilde{y}^{k-1}\right)\right\| \\ &\quad+\left\|F^{k}\left(\tilde{y}^{k-1},\tilde{y}^{k-2}\right)-F\left(\tilde{y}^{k-1},\tilde{y}^{k-2}\right)\right\| \\ &\quad+\left\|F^{k}\left(\tilde{y}^{k},\tilde{y}^{k-1}\right)-F\left(\tilde{y}^{k-1},\tilde{y}^{k-2}\right)\right\|. \qquad (92)\end{aligned}$$

Given that $\left\|F^{k}(y)-F(y)\right\| \le \delta^{k}$ for all $y \in Y$ then

$$\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\| \le \delta^{k+1}+\delta^{k}+\left\|F\left(\tilde{y}^{k},\tilde{y}^{k-1}\right)-F\left(\tilde{y}^{k-1},\tilde{y}^{k-2}\right)\right\|. \qquad (93)$$

By the contraction property of F,

$$\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\|_{b} \le \delta^{k+1}+\delta^{k}+\alpha_1\left\|\tilde{y}^{k}-\tilde{y}^{k-1}\right\|_{b}+\alpha_2\left\|\tilde{y}^{k-1}-\tilde{y}^{k-2}\right\|_{b}. \qquad (94)$$

Recursively substituting yields

$$\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\|_{b} \le \sum_{j=0}^{k+1} C_j\left(\delta^{k+1-j}+\delta^{k-j}\right), \qquad (95)$$

where $C_j \triangleq \alpha_1 C_{j-1}+\alpha_2 C_{j-2}$, and $C_0=1$, $C_1=\alpha_1$. If $\delta^{k}<\delta$ for all k then

$$\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\|_{b} \le 2\,\delta\sum_{j=0}^{k+1} C_j, \qquad (96)$$

where

$$\sum_{j=0}^{k+1} C_j = \sum_{J=0}^{k/2}\alpha_2^{J}\sum_{I=0}^{k-2J}\alpha_1^{I}\cdot C_{JI} \qquad (97)$$

and

$$C_{JI} = \underbrace{\sum_{i_J=0}^{I}\sum_{i_{J-1}=0}^{i_J}\cdots\sum_{i_1=0}^{i_2} 1}_{J}. \qquad (98)$$

Taking the limit as $k\to\infty$ yields

$$\lim_{k\to\infty}\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\| \le \frac{2\delta}{1-(\alpha_1+\alpha_2)}. \qquad (99)$$

Since $(\alpha_1+\alpha_2)<1$, the $\lim_{k\to\infty}\left\|\tilde{y}^{k+1}-\tilde{y}^{k}\right\|$ can be made smaller than any desired ϵ by choosing $\delta(\epsilon)<(\epsilon/2)(1-(\alpha_1+\alpha_2))$, thus proving the first part of the theorem.

Let y^{*} be the fixed point of F. The difference between the $(k+1)^{\text{st}}$ iterate and the exact solution becomes

$$\left\|\tilde{y}^{k+1}-y^{*}\right\| = \left\|F^{k+1}\left(\tilde{y}^{k},\tilde{y}^{k-1}\right)-F\left(y^{*},y^{*}\right)\right\|. \qquad (100)$$

Following the previous construction, this becomes

$$\left\|\tilde{y}^{k+1}-y^{*}\right\|_{b} \le \delta^{k+1}+\alpha_1\left\|\tilde{y}^{k}-y^{*}\right\|_{b}+\alpha_2\left\|\tilde{y}^{k-1}-y^{*}\right\|_{b}. \qquad (101)$$

Recursively substituting and taking the limit yields

$$\lim_{k\to\infty}\left\|\tilde{y}^{k+1}-y^{*}\right\|_{b} \le \frac{\delta}{1-(\alpha_1+\alpha_2)}, \qquad (102)$$

which completes the proof of the theorem. ∎

REFERENCES

1. D.J. Hill and I.M.Y. Mareels, Stability theory for differential/algebraic systems with applications to power systems, *IEEE Transactions on Circuits and Systems* **37**, 1416–1423 (November 1990).
2. G.C. Verghese, B.C. Levy and T. Kailath, A generalized state-space for singular systems, *IEEE Transactions on Automatic Control*, 811–831 (August 1981).
3. P.V. Kokotovic, R.E. O'Malley, Jr. and P. Sannuti, Singular perturbations and order reduction in control theory—An overview, *Automatica* **12**, 123–132 (March 1976).
4. S.L. Campbell, The numerical solution of singular systems arising in control problems, *Proceedings of the 1989 American Control Conference*, Pittsburgh, PA, (June 1989).
5. R.F. Sincovec, B. Dembart, M.A. Epton, A.M. Erisman, J.W. Manke and E.L. Yip, Solvability of large-scale descriptor systems, Report, Boeing Computer Services Company, Seattle, WA, (1979).
6. M.W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, John Wiley and Sons, New York, (1989).
7. K.E. Brenan, S.L. Campbell and L.R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, Elsevier Science Publishing Co., New York, (1989).
8. S.L. Campbell, The numerical solution of higher index linear time varying singular systems of differential equations, *SIAM Journal of Scientific and Statistical Computing* **6** (2), 334–348 (April 1985).
9. C.W. Gear and L.R. Petzold, ODE methods for the solution of differential/algebraic systems, *SIAM Journal of Numerical Analysis* **21** (4), 716–728 (August 1984).
10. C.W. Gear, G.K. Gupta and B. Leimkuhler, Automatic integration of Euler-Lagrange equations with constraints, *Journal of Computational and Applied Mathematics* **12/13**, 77–90 (1985).
11. B.J. Leimkuhler, Error estimates for differential-algebraic equations, Master's Thesis, University of Illinois, Urbana, IL, (May 1986).
12. B.J. Leimkuhler, Approximation methods for the consistent initialization of differential-algebraic equations, Report No. UIUCDCS-R-88-1450, University of Illinois, Urbana, IL, (August 1988).
13. P. Lötstedt and L. Petzold, Numerical solution of nonlinear differential equations with algebraic constraints I: Convergence results for backward differentiation formulas, *Mathematics of Computation* **46** (174), 491–516 (April 1986).
14. I.M. Mack, Block implicit one-step methods for solving smooth and discontinuous systems of differential/algebraic equation, Ph.D. Dissertation, Harvard University, Cambridge, MA, (1986).
15. L. Petzold and P. Lötstedt, Numerical solution of nonlinear differential equations with algebraic constraints II: Practical implications, *SIAM Journal of Scientific Statistical Computation* **7** (3), 720–733 (July 1986).
16. L.O. Chua and P. Lin, *Computer Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques*, Prentice-Hall, Englewood Cliffs, NJ, (1975).
17. J.K. White and A. Sangiovanni-Vincentelli, *Relaxation Techniques for the Simulation of VLSI Circuits*, Kluwer Academic Publishers, Boston, MA, (1987).
18. J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, (1970).
19. E. Lelarasmee, A.E. Ruehli and A.L. Sangiovanni-Vincentelli, The waveform relaxation method for time-domain analysis of large scale integrated circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **CAD-1** (3), 131–144 (July 1982).
20. O. Nevanlinna and F. Odeh, Remarks on the convergence of the waveform relaxation method, *Numer. Funct. Anal. Optimiz.* **9**, 435–445 (1987).
21. M.L. Crow, Waveform relaxation methods for the simulation of systems of differential/algebraic equations with applications to electric power systems, Ph.D. Dissertation, University of Illinois, Urbana, IL, (January 1990).
22. O. Nevanlinna, Remarks on Picard-Lindelöf iteration: Part I, *BIT* **29**, 328–346 (1988).
23. O. Nevanlinna, Remarks on Picard-Lindelöf iteration: Part II, *BIT* **29**, 535–562 (1988).