INITIAL CONDITION CALCULATION

C. A. GEORGE
General Electric Company
Utica, New York

ABSTRACT

A method is presented for the calculation of the initial values of the dependent variables and their derivatives when the network equations are mixed differential-algebraic and implicit in the derivatives. That is, it calculates values for \( W(0) \) and \( W'(0) \) for systems of equations in the form \( q(W,W,t) = 0 \). The crux of the problem can be revealed as follows: Equations in the above form do not reveal a dynamically independent set of variables. Even if a set is known and initial values are assigned to them, there still remains the need to calculate the initial values for the remaining variables and for the derivatives of all the variables. This is true even for derivatives that do not appear in the network equations since knowledge of their values simplifies integration of the network equations by a predictor-corrector scheme. The method presented allows initial conditions to be specified by a system of equations of the form \( p(W(0),W'(0)) = 0 \). (This corresponds to specifying the values of the dynamic variables in some other methods). If the number of arbitrary initial conditions allowable by \( q \) is greater than specified by \( p \), the method will set a sufficient number of elements of \( W \) and/or \( W' \) to zero.

1. INTRODUCTION

When network equations are in the form:

\[
\begin{align*}
\dot{w} &= f(w,m,t) \\
0 &= g(w,m,t)
\end{align*}
\]

initial conditions can be defined by specifying the value of \( w(0) \). The values of \( m \) and \( \dot{w} \) can then be calculated. There are other network equation formulations which do not lend themselves to such a simple determination of which variables are dynamically independent. This paper relates to programs which use formulations in which a set of dynamically independent variables is not always evident and to the specifying of initial conditions in a convenient manner. (Even if independent dynamic variables are known, they may not be directly related to the physics of the problem in a manner that makes them convenient to specify.)

A simple example of the types of formulations being considered is given by the following equations (which apply at \( t=0 \)):

\[
\begin{align*}
0 &= \dot{w}_1 - w_2 - w_4 \\
0 &= \dot{w}_2 - w_1 w_2 \\
0 &= w_4 - w_2 + w_1 w_2 \\
0 &= w_1 - w_3 - w_2 - w_4.
\end{align*}
\]

If \( w_1(0) = w_2(0) = 0 \), then the following can be calculated: \( w_3(0) = \dot{w}_2(0) = 0 \) and \( w_4(0) = \dot{w}_1(0) = 1 \). Furthermore, if the equations apply to a region about \( t=0 \) then \( \dot{w}_4(0) = 0 \) and \( \dot{w}_3(0) = 1 \). This is sufficient information to predict \( W(t+h) \) so that a predictor-corrector which solves systems differential-algebraic equations can be applied.

The driving force for this analysis relates to a program, BELAC[1], which allows the user to define multiterminal elements with subroutines which calculate the voltages and/or current at the elements' terminals as functions of other voltages, currents and derivatives. The network equation generating routine knows only the interconnection of...
The scheme described in the sequel can be used to calculate the initial values of the dependent variables and their derivatives when the network equations are mixed differential-algebraic and implicit in the derivatives. That is, it calculates values for \( \dot{W}(0) \) and \( \ddot{W}(0) \) for systems of equations in the form \( q(W,\dot{W},t) = 0. \) The crux of the problem can be revealed as follows: Equations in the above form do not reveal a dynamically independent set of variables. Even if a set is known and initial values are assigned to them, there still remains the need to calculate the initial values for the remaining variables and for the derivatives of all the variables. This is true even for derivatives that do not appear in the network equations since knowledge of their values simplifies integration of the network equations by a predictor-corrector scheme. The method presented allows initial conditions to be specified by a system of equations of the form \( p(W(0),\dot{W}(0)) = 0. \) (This corresponds to specifying values of dynamic variables in some other methods). If the number of arbitrary initial conditions allowable by \( q \) is greater than specified by \( p \) the method will set a sufficient number of elements of \( W \) and/or \( \dot{W} \) to zero.

3. DEVIATION

Let the network equations be:

\[
q(W,\dot{W},t) = 0. \tag{3}
\]

Let the user supplied initial conditions (if any) be given by:

\[
p(W(0),\dot{W}(0)) = 0. \tag{4}
\]

The equations in Eq. 4 need not be independent nor independent of the network equations. Usually Eq. 3 and Eq. 4 combined (at \( t=0 \)) will not be sufficient to allow the complete determination of \( W(0) \) and \( \dot{W}(0) \). This is remedied by using:

\[
\dot{q}(W(0),\dot{W}(0),0) = 0. \tag{5}
\]

In what follows \( q_X \) is used to indicate the partial derivative of \( q \) with respect to \( W \), \( q_Y \) the partial derivative with respect to \( \dot{W} \), and \( q_T \) the partial derivative with respect to time; all evaluated at \( t=0, W(0), \dot{W}(0) \). Similar definitions apply to \( p_X \) and \( p_Y \). Using this notation, Eq. 5 can be expanded to:

\[
\dot{q}(W(0),\dot{W}(0),0) = q_XW(0) + q_Y\dot{W}(0) + q_T = 0. \tag{6}
\]

Expanding Eq. 3 and 4 in Taylor series about \( (W_0,\dot{W}_0,0) \) and dropping the high order terms, and using elementary transformations on Eq. 6 gives:
The equations containing $\dot{W}(0)$ are not needed and will, therefore, be eliminated from further considerations.

If Eq. 3 or 4 is non-linear, Eq. 7 is a linear approximation that will be used in a Newton iteration scheme. It is, therefore, convenient to define $Z=W(0)-W_0$ and $\dot{Z}=\dot{W}(0)-\dot{W}_0$. Eq. 7 then becomes:

$$
\begin{bmatrix}
q_x & q_y \\
p_x & p_y \\
0 & AB
\end{bmatrix}
\begin{bmatrix}
Z \\
\dot{Z}
\end{bmatrix} =
\begin{bmatrix}
-q(W_0,\dot{W}_0,0) \\
-p(W_0,\dot{W}_0)
\end{bmatrix}
$$

which is solved for $Z$ and $\dot{Z}$. Then $W_0$ and $\dot{W}_0$ is replaced by $W_0+Z$ and $\dot{W}_0+\dot{Z}$, respectively, and another iteration performed if necessary.

The plausibility of Eq. 8 can be increased by considering that if there are $n$ dependent variables, the rank of the topmost set is $n$. If the number of dynamically independent variables is $m$, the rank of the bottom-most set of equations is $n-m$. Since there are $2n$ dependent variables to be determined ($W$ and $\dot{W}$) there remains $2n-n-(n-m)=m$ initial conditions that may be specified by the center set of equations, as one would expect.

Care must be exercised in solving for $Z$ and $\dot{Z}$ since the user supplied set of equations may not be independent of the other sets of equations in Eq. 8. Furthermore, the rank of Eq. 8 may be less than $2n$, in which case some values in $Z$ and/or $\dot{Z}$ may be arbitrarily assigned (zero seems to be the obvious choice).

To solve Eq. 8 define $a$, $b$ and $c$ such that:

$$
az+b\dot{z}-c=0
$$

By performing Gauss row reduction on $\mathbf{a}$ (with row and column interchange if necessary) Eq. 9 can be put into the following form:

$$
\begin{bmatrix}
U & a_1 \\
b_1 & b_2
\end{bmatrix}
\begin{bmatrix}
Z_a+ \\
\dot{Z}_a=c_a.
\end{bmatrix}
$$

The subscripts on $Z_a$ and $\dot{Z}_a$ indicate that the operations may have reordered the elements of $Z$ and $\dot{Z}$. $U$ is a unit matrix. By performing Gauss' row reduction on $b_4$ (with row and column interchange if necessary) Eq. 10 can be put into the following form.

$$
\begin{bmatrix}
U & a_2 \\
b_5 & b_6
\end{bmatrix}
\begin{bmatrix}
Z_b+ \\
\dot{Z}_b=c_b
\end{bmatrix}
$$

By performing Gauss row reduction on $b_9$ (with row and column interchanges if necessary) Eq. 11 can be put into the following form:

$$
\begin{bmatrix}
U & a_4 & a_5 \\
b_1 & b_0 & b_1
\end{bmatrix}
\begin{bmatrix}
Z_1 \\
\dot{Z}_1
\end{bmatrix} = c_1
$$

$$
\begin{bmatrix}
0 & a_6 & a_7 \\
b_2 & b_0 & b_3
\end{bmatrix}
\begin{bmatrix}
Z_2 \\
\dot{Z}_2
\end{bmatrix} = c_2
$$

$$
\begin{bmatrix}
0 & 0 & 0 \\
b_3 & b_0 & b_3
\end{bmatrix}
\begin{bmatrix}
Z_3 \\
\dot{Z}_3
\end{bmatrix} = c_3
$$

$$
\begin{bmatrix}
0 & 0 & 0 \\
b_4 & b_0 & b_4
\end{bmatrix}
\begin{bmatrix}
Z_4 \\
\dot{Z}_4
\end{bmatrix} = c_4
$$

$$
\begin{bmatrix}
0 & 0 & 0 \\
b_5 & b_0 & b_5
\end{bmatrix}
\begin{bmatrix}
Z_5 \\
\dot{Z}_5
\end{bmatrix} = c_5
$$

If two columns are interchanged during the reduction of $b_9$, the corresponding rows are also interchanged so that the unit matrix in the coefficient matrix of $Z_b$ will remain
diagonal. Of course the partitioning of \( Z \) and \( \dot{Z} \) must be such as to make them conformal with multiplications indicated and is shown explicitly in Eq. 12. The sizes of the partitions shown in Eq. 12 are determined by the various Gauss reduction steps and are indicated by the variables \( I_1, I_2, \) and \( I_3 \) which are defined by the size of the unit matrices as follows:

\[
\begin{align*}
U_1 & = I_1 \times I_1, \\
U_2 & = I_2 \times I_2, \\
U_3 & = I_3 \times I_3, \text{ and} \\
U_4 & = I_1 \times I_1.
\end{align*}
\]

Arbitrarily set \( \dot{Z}_2 = \dot{Z}_4 = Z_3 = Z_4 = 0. \) This is done, as was this entire analysis, with an eye on what is to be accomplished and based on insight. The result of these assignments is that a solution is evidently:

\[
\begin{align*}
Z_1 &= c_1, \\
Z_2 &= c_2, \\
\dot{Z}_3 &= c_3, \text{ and} \\
\dot{Z}_1 &= c_4.
\end{align*}
\]

or,

\[
Z = \begin{bmatrix} c_1 \\ c_2 \\ 0 \\ 0 \end{bmatrix}, \text{ and } \dot{Z} = \begin{bmatrix} c_4 \\ 0 \\ c_3 \\ 0 \end{bmatrix}
\]

Finally, setting replacing \( \dot{W} \) and \( \dot{W} \) with \( W_0 + Z \) and \( \dot{W}_0 + \dot{Z} \) completes one iteration.

In some networks one or more of the unit matrices will not be needed. For example, \( I_2 \) may be zero. In this case Eq. 12 reduces to a form which does not have the second row and column shown in the coefficient matrices and \( c_2, Z_2 \) and \( \dot{Z}_2 \) are also nonexistent.

If the initial condition equations are inconsistent with the network equations \( c_5 \) in Eq. 12 will not be zero. This can be tested by any digital computer program using this algorithm. However, if floating point arithmetic is used, round-off errors could produce "small" numbers in \( c_5 \) even with no inconsistencies.

This algorithm does not require the initial condition equations to be independent of the network equations. For example, an initial condition equation might require \( W_1(0) - W_2(0) = 0 \), while a network equation may specify that \( W_1 = W_2 \) for all time. Such extra initial condition equations produce the 0=0 equations associated with \( c_5 \). The attribute of allowing extra initial condition equations is desirable since the network equations are generated by the computer, not the user, while Eq. 4 is created by the user. Since the user does not know what is contained in the network equations, he could easily specify initial-condition equations that are not independent of them.

The derivation above trianglized the coefficient matrix of \( Z \) and then performed operations on the coefficient matrix of \( \dot{Z} \). This was done so that the resultant algorithm would choose to update \( \dot{W} \) rather than \( \dot{W} \) whenever it had a choice. This is consistent with the need for \( \dot{W} \) being zero at quiescence.
REFERENCES

