SIAM J. Sci. Comput.
Vol. 24, No. 3, pp. 1076–1089

ADJOINT SENSITIVITY ANALYSIS FOR DIFFERENTIAL-ALGEBRAIC EQUATIONS: THE ADJOINT DAE SYSTEM AND ITS NUMERICAL SOLUTION*

YANG CAO†, SHENGTAI LI‡, LINDA PETZOLD†, AND RADU SERBAN§

Abstract. An adjoint sensitivity method is presented for parameter-dependent differential-algebraic equation systems (DAEs). The adjoint system is derived, along with conditions for its consistent initialization, for DAEs of index up to two (Hessenberg). For stable linear DAEs, stability of the adjoint system (for semi-explicit DAEs) or of an augmented adjoint system (for fully implicit DAEs) is shown. In addition, it is shown for these systems that numerical stability is maintained for the adjoint system or for the augmented adjoint system.

Key words. sensitivity analysis, DAE, adjoint method

AMS subject classifications. 65L10, 65L99

PII. S1064827501380630

1. Introduction.

With the rapid development of faster computers, increasingly realistic mathematical models are being used to investigate physical phenomena. New model features often call for parameters whose values may not be accurately known. Thus there is a need for parametric sensitivity analysis of differential-algebraic models. Areas of application include optimization, parameter estimation, model simplification, data assimilation, optimal control, process sensitivity, uncertainty analysis, and experimental design for a wide range of scientific and engineering problems.

Recent work on methods and software for sensitivity analysis of DAE systems [15, 22, 20, 21, 23] has demonstrated that forward sensitivities can be computed reliably and efficiently via automatic differentiation [8] in combination with DAE solution techniques designed to exploit the structure of the sensitivity system. The DASPK3.0 [20, 21] software package was developed for forward sensitivity analysis of DAE systems of index up to two and has been used in sensitivity analysis and design optimization of several large-scale engineering problems [19, 25]. DASPK3.0 is an extension of the DASPK software [9, 10] developed by Brown, Hindmarsh, and Petzold for the solution of large-scale DAE systems. For a parameter-dependent DAE system

$$\begin{cases} F(x, \dot{x}, t, p) = 0, \\ \qquad x(0) = x_0(p), \end{cases} \tag{1}$$

where $x \in R^{n_x}$ and $p \in R^{n_p}$, these problems take the following form: find dx/dp_j

*Received by the editors September 3, 2001; accepted for publication (in revised form) June 17, 2002; published electronically January 23, 2003. This work was supported by grants EPRI WO-8333-06, NSF CCR 98-96198, NSF/ARPA PC 239415, NSF/KDI ATM-9873133, and LLNL ISCR 00-15.

http://www.siam.org/journals/sisc/24-3/38063.html

†Department of Computer Science, University of California, Santa Barbara, CA 93106 (ycao@engineering.ucsb.edu, petzold@engineering.ucsb.edu).

‡Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545 (shenli@t7.lanl.gov).

§Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551 (radu@llnl.gov). The work of the last author was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory, under contract W-7405-Eng-48.

at time T, for $j = 1, \ldots, n_p$. Their solution requires the simultaneous solution of the original DAE system with the n_p sensitivity systems obtained by differentiating the original DAE with respect to each parameter in turn. For large systems this may look like a lot of work but it can be done efficiently, if n_p is relatively small, by exploiting the fact that the sensitivity systems are linear and all share the same Jacobian matrices with the original system.

Some problems require the sensitivities with respect to a large number of parameters. For these problems, particularly if the number of state variables is also large, the forward sensitivity approach is intractable. These problems can often be handled more efficiently by the adjoint method [14]. In this approach, we are interested in calculating the sensitivity $\frac{dG}{dp}$ of an objective function

$$G(x,p) = \int_0^T g(x,t,p)dt, \tag{2}$$

or alternatively the sensitivity $\frac{dg}{dp}$ of a function $g(x,T,p)$ defined only at time T. The function g must be smooth enough that g_p and g_x exist and are bounded. The primary cost in computing $\frac{dG}{dp}$ or $\frac{dg}{dp}$ via the adjoint method is the calculation of the intermediate quantity λ, called the adjoint variable, as the solution of the adjoint system. The adjoint system is a linear DAE system associated with the governing DAEs (1). While forward sensitivity analysis is best suited to the situation of finding the sensitivities of a potentially large number of solution variables with respect to a small number of parameters, reverse (adjoint) sensitivity analysis is best suited to the complementary situation of finding the sensitivity of a scalar (or small-dimensional) function of the solution with respect to a large number of parameters.

The specification of the adjoint system for DAEs in the absence of sensitivity considerations is straightforward. For a linear DAE of the form

$$A\dot{x} + Bx = 0, \tag{3}$$

where A and B are sufficiently smooth matrix functions, the adjoint or *dual* DAE is given by

$$(A^*\lambda)' - B^*\lambda = 0, \tag{4}$$

where $*$ denotes the conjugate transpose and $'$ denotes the time derivative. In related works [3, 4, 5, 7, 6], Balla and März have derived the adjoint system for index-1 DAEs and showed its solvability and index-1 structure under minimal conditions of smoothness. When the adjoint method is used for computation of the sensitivities, the adjoint system (4) may become inhomogeneous and have different initial conditions, depending on the form of the objective functions. Our contributions in the present paper are first to give a simple derivation using variational methods of the adjoint system for general nonlinear DAEs and to show how to use it in sensitivity analysis. Then we present conditions for the consistent initialization of the adjoint DAE system for DAEs of index up to two (Hessenberg). Finally, we show for linear DAEs that if the original DAE is stable, then the adjoint DAE (for semi-explicit DAEs) or an augmented adjoint DAE (for fully implicit DAEs) is stable, and that numerical stability is maintained for the adjoint DAE or the augmented adjoint DAE.

We note that März [24] has shown that for "well-formulated" DAE systems (these DAE systems must be written in a specific form), the operations of adjoint and discretization commute. For such systems, stability of the adjoint and its discretization is maintained, hence there is no need for an augmented adjoint DAE. There are many computational advantages for such DAE systems, hence we advocate rewriting the system into this form whenever possible. However, for large nonlinear problems from scientific computing this may not always be possible or convenient. Thus we believe it is important also to consider numerical methods for DAEs of the more general form (1).

The outline of this paper is as follows. In section 2 we derive the adjoint system by a variational method, along with conditions for its consistent initialization for index-0 and index-1 DAEs. In section 3 we give some examples and derive the conditions for the consistent initialization of the adjoint system for Hessenberg index-2 DAEs. In section 4 we investigate the stability of the adjoint system and show how to formulate an augmented adjoint system for which stability is maintained even for fully implicit DAE systems. In section 5 we address the numerical stability of the adjoint and augmented adjoint DAE systems.

2. Derivation of the adjoint system for sensitivity.

2.1. Sensitivity of $G(x, p)$. We focus first on solving the sensitivity problem for $G(x, p)$ defined by (2). Introducing a Lagrange multiplier λ, we form the augmented objective function

$$I(x,p) = G(x,p) - \int_0^T \lambda^* F(x, \dot{x}, t, p) dt.$$

Since $F(x, \dot{x}, t, p) = 0$, the sensitivity of G with respect to p is

$$\frac{dG}{dp} = \frac{dI}{dp} = \int_0^T (g_p + g_x x_p) dt - \int_0^T \lambda^* (F_p + F_x x_p + F_{\dot{x}} \dot{x}_p) dt, \tag{5}$$

where subscripts on functions such as F or g are used to denote partial derivatives. By integration by parts, we have

$$\int_0^T \lambda^* F_{\dot{x}} \dot{x}_p dt = (\lambda^* F_{\dot{x}} x_p)|_0^T - \int_0^T (\lambda^* F_{\dot{x}})' x_p dt.$$

Thus (5) becomes

$$\frac{dG}{dp} = \int_0^T (g_p - \lambda^* F_p)\, dt - \int_0^T \left[-g_x + \lambda^* F_x - (\lambda^* F_{\dot{x}})' \right] x_p dt - (\lambda^* F_{\dot{x}} x_p)|_0^T. \tag{6}$$

Now letting

$$(\lambda^* F_{\dot{x}})' - \lambda^* F_x = -g_x, \tag{7}$$

we obtain

$$\frac{dG}{dp} = \int_0^T (g_p - \lambda^* F_p)\, dt - (\lambda^* F_{\dot{x}} x_p)|_0^T. \tag{8}$$

Note that x_p at $t = 0$ is the sensitivity of the initial conditions with respect to p, which is easily obtained. To find the initial conditions (at $t = T$) for the adjoint system, we must take into consideration the structure of the DAE system.

For index-0 and index-1 DAE systems, we can simply take

$$\lambda^* F_{\dot{x}}|_{t=T} = 0, \tag{9}$$

yielding the sensitivity equation for $\frac{dG}{dp}$

$$\frac{dG}{dp} = \int_0^T (g_p - \lambda^* F_p)\, dt + (\lambda^* F_{\dot{x}} x_p)|_{t=0}. \tag{10}$$

This choice will not suffice for a Hessenberg index-2 DAE system. Consider the system

$$\begin{cases} \dot{x}^d = f^1(x^d, x^a, p), \\ \;\; 0 = f^2(x^d, p), \end{cases} \tag{11}$$

where x^d and x^a denote the differential and algebraic solution components, respectively, $A = \frac{\partial f^1}{\partial x^d}$, $B = \frac{\partial f^1}{\partial x^a}$, $C = \frac{\partial f^2}{\partial x^d}$, and CB invertible. Here the adjoint system is given by

$$\begin{cases} \dot{\lambda}^{d*} + \lambda^{d*} A + \lambda^{a*} C = -g_{x^d}, \\ \qquad\qquad\quad \lambda^{d*} B = -g_{x^a}, \end{cases} \tag{12}$$

where λ^d and λ^a denote the differential and algebraic adjoint variables, respectively. If the initial conditions are set as in (9), then $\lambda^d(T) = 0$, which may be in conflict with the constraint equations of (12) if $g(x, p)$ depends explicitly on x^a. To resolve this potential conflict, we require

$$\lambda^{d*}(T) = \xi^* C|_{t=T}, \tag{13}$$

where ξ^* is yet to be determined. Because $f^2(x^d, p) = 0$, we have $Cx_p^d = -f_p^2$. Thus if λ^d satisfies (13), we have

$$\lambda^{d*} x_p^d|_{t=T} = -\xi^* f_p^2|_{t=T}. \tag{14}$$

Inserting (13) into the constraint equation (12), we have

$$\xi^* CB|_{t=T} = -g_{x^a}|_{t=T}.$$

Since CB is invertible, $\xi^* = -g_{x^a}(CB)^{-1}|_{t=T}$, and the boundary condition for λ^d is

$$\lambda^{d*}|_{t=T} = -g_{x^a}(CB)^{-1} C|_{t=T}. \tag{15}$$

The sensitivity equation (8) then becomes

$$\frac{dG}{dp} = \int_0^T \left(g_p + \lambda^{d*} f_p^1 + \lambda^{a*} f_p^2\right) dt + (\lambda^{d*} x_p^d)|_{t=0} + g_{x^a}(CB)^{-1} f_p^2. \tag{16}$$

Thus we have derived the sensitivity equation for $\frac{dG}{dp}$ along with the adjoint DAE system for λ and its boundary condition at $t = T$ for index-0, index-1, and index-2 Hessenberg DAE systems.

2.2. Sensitivity of $g(x, T, p)$. Now let us consider the computation of $\frac{dg}{dp}$. From $\frac{dg}{dp} = \frac{d}{dT}\frac{dG}{dp}$ and (8), we have

$$\frac{dg}{dp} = (g_p - \lambda^* F_p)(T) - \int_0^T \lambda_T^* F_p dt + (\lambda_T^* F_{\dot{x}} x_p)|_{t=0} - \frac{d(\lambda^* F_{\dot{x}} x_p)}{dT}, \tag{17}$$

where λ_T denotes $\frac{\partial \lambda}{\partial T}$. For index-0 and index-1 DAEs, $\frac{d(\lambda^* F_{\dot{x}} x_p)|_{t=T}}{dT} = 0$. For a Hessenberg index-2 DAE system of the form (11),

$$\frac{d(\lambda^* F_{\dot{x}} x_p)|_{t=T}}{dT} = -\left.\frac{d(g_{x^a}(CB)^{-1} f_p^2)}{dt}\right|_{t=T}.$$

The corresponding adjoint equations are

$$(\lambda_T^* F_{\dot{x}})' - \lambda_T^* F_x = 0. \tag{18}$$

For index-0 and index-1 DAEs (as shown above, the index-2 case is different), to find the boundary condition for this equation we write λ as $\lambda(t, T)$ because it depends on both t and T. Then

$$\lambda^*(T, T) F_{\dot{x}}|_{t=T} = 0.$$

Taking the total derivative, we obtain

$$(\lambda_t + \lambda_T)^*(T, T) F_{\dot{x}}|_{t=T} + \lambda^*(T, T) \frac{dF_{\dot{x}}}{dt} = 0.$$

Since λ_t is just $\dot{\lambda}$, we have the boundary condition

$$(\lambda_T^* F_{\dot{x}})|_{t=T} = -\left.\left[\lambda^*(T, T) \frac{dF_{\dot{x}}}{dt} + \dot{\lambda}^* F_{\dot{x}}\right]\right|_{t=T}. \tag{19}$$

For the index-one DAE case, (7) and (19) yield

$$(\lambda_T^* F_{\dot{x}})|_{t=T} = [g_x - \lambda^* F_x]\,|_{t=T}. \tag{20}$$

For the regular implicit ODE case, $F_{\dot{x}}$ is invertible; thus we have $\lambda(T, T) = 0$, which leads to $\lambda_T(T) = -\dot{\lambda}(T)$.

We will discuss the appropriate boundary condition for index-2 Hessenberg DAEs in the next section.

3. Examples.

For simplicity of presentation, throughout this section we assume that the only dependency of the differential equations on the parameters p is through the initial conditions and that the objective function g does not depend explicitly on p. This is often the case in applications. We also assume that the initial conditions are consistent with the algebraic constraints (including any hidden constraints) in a neighborhood of the nominal values of the parameters.

3.1. Standard form ODE.

Given the ODE initial value problem

$$\begin{cases} \dot{x} = f(x, t), \\ x(0) = x_0(p) \end{cases}$$

and the function $g(x)$ at $t = T$, the corresponding adjoint system is

$$\begin{cases} \dot{\lambda}_T = -f_x^* \lambda_T, \\ \lambda_T(T) = g_x^* \end{cases}$$

and the sensitivity is given by $\frac{dg}{dp} = \lambda_T^*(0) x_{0p}$, where $x_{0p} = \frac{dx_0}{dp}$. This equation agrees with the adjoint system as commonly defined for ODEs.

3.2. Implicit ODE. Given

$$\begin{cases} F(\dot{x}, x, t) = 0, \\ \quad x(0) = x_0(p), \end{cases}$$

with $A = \frac{\partial F}{\partial \dot{x}}$ nonsingular, $B = \frac{\partial F}{\partial x}$, and the function $g(x)$ at $t = T$, the corresponding adjoint system is

$$\begin{cases} (A^* \lambda_T)' - B^* \lambda_T = 0, \\ \quad A^* \lambda_T(T) = g_x^* \end{cases}$$

and the sensitivity is given by $\frac{dg}{dp} = \lambda_T^*(0) A(0) x_{0p}$.

3.3. Semi-explicit index-1 DAE. Given

$$\begin{cases} \dot{x} = f^1(x^d, x^a, t), \\ 0 = f^2(x^d, x^a), \\ x^d(0) = x_0^d(p), \end{cases}$$

with $A = \frac{\partial f^1}{\partial x^d}$, $B = \frac{\partial f^1}{\partial x^a}$, $C = \frac{\partial f^2}{\partial x^d}$, $D = \frac{\partial f^2}{\partial x^a}$ nonsingular, and the function $g(x^d)$ at $t = T$, the corresponding adjoint system is

$$\begin{cases} \dot{\lambda}_T^d = -A^* \lambda_T^d - C^* \lambda_T^a, \\ 0 = B^* \lambda_T^d + D^* \lambda_T^a, \\ \lambda_T^d(T) = g_{x^d}^* \end{cases}$$

and the sensitivity is given by $\frac{dg}{dp} = \lambda_T^{d*}(0) \frac{dx_0^d}{dp}$.

If the function g depends on both x^d and x^a, $g = g(x^d(T), x^a(T))$, the adjoint equations are the same as above, but the boundary condition is now given by

$$\lambda_T^d(T) = g_{x^d}^* - C^*(D^*)^{-1} g_{x^a}^*.$$

3.4. Hessenberg index-2 DAE. Given

$$\begin{cases} \dot{x}^d = f^1(x^d, x^a, t), \\ 0 = f^2(x^d), \\ x^d(0) = x_0^d(p), \end{cases}$$

with $A = \frac{\partial f^1}{\partial x^d}$, $B = \frac{\partial f^1}{\partial x^a}$, $C = \frac{\partial f^2}{\partial x^d}$, and CB invertible, and given that the function $g(x, T, p)$ depends only on the differential components x^d of x, the corresponding adjoint system is

$$\begin{cases} \dot{\lambda}_T^d = -A^* \lambda_T^d - C^* \lambda_T^a, \\ 0 = B^* \lambda_T^d, \\ \lambda_T^d(T) = (I - C^*(B^* C^*)^{-1} B^*) g_{x^d}^* \end{cases}$$

and the sensitivity is given by $\frac{dg}{dp} = \lambda_T^{d*}(0) \frac{dx_0^d}{dp}$.

If $g(x, T, p)$ depends on both x^d and x^a, we must solve for the boundary condition first, from

$$\text{(21)} \qquad \begin{cases} \dot{\lambda}^d + A^* \lambda^d + C^* \lambda^a = -g_{x^d}^*, \\ 0 = B^* \lambda^d + g_{x^a}^*, \\ \lambda^d(T) = -C^*(B^* C^*)^{-1} g_{x^a}^*. \end{cases}$$

The adjoint system is

$$\begin{cases} \dot{\lambda}^d_T = -A^*\lambda^d_T - C^*\lambda^a_T, \\ 0 = B^*\lambda^d_T. \end{cases}$$

As in (15), we have $\lambda^{d*}|_{t=T} = -g_{x^a}|_{t=T}(CB)^{-1}C$. If B and C are constant, we can take the total derivative to obtain

$$(\lambda^d_t + \lambda^d_T)^*(T,T) = -\left.\frac{dg_{x^a}}{dt}\right|_{t=T}(CB)^{-1}C,$$

so that

$$\lambda^{d*}_T(T,T) = -\dot{\lambda}^{d*}|_{t=T} - \left.\frac{dg_{x^a}}{dt}\right|_{t=T}(CB)^{-1}C.$$

Substituting for $\dot{\lambda}^d$ from (21), we obtain

$$\lambda^d_T(T,T) = (I - C^*(B^*C^*)^{-1}B^*)\left[g^*_{x^d}|_{t=T} + A^*\lambda^d(T,T)\right],$$

where $\lambda^d(T,T) = -C^*(B^*C^*)^{-1}g^*_{x^a}|_{t=T}$. Denoting $P = I - B(CB)^{-1}C$, $\lambda^d_T(T,T)$ can be expressed as

$$\lambda^d_T(T,T) = P^*(g^*_{x^d} + A^*\lambda^d(T,T)).$$

If B and C are not constant, we take the derivative of

$$\lambda^{d*}|_{t=T} = -g_{x^a}(CB)^{-1}C|_{t=T}$$

to obtain

$$(\lambda^d_t + \lambda^d_T)^*(T,T) = -\left.\frac{dg_{x^a}}{dt}\right|_{t=T}(CB)^{-1}C - g_{x^a}|_{t=T}\frac{d((CB)^{-1}C)}{dt}$$

or

$$\lambda^d_T|_{t=T} = -\dot{\lambda}^d - C^*(B^*C^*)^{-1}\left.\frac{dg^*_{x^a}}{dt}\right|_{t=T} - \left.\frac{d(C^*(B^*C^*)^{-1})}{dt}g^*_{x^a}\right|_{t=T}.$$

From

$$\frac{d(C^*(B^*C^*)^{-1})}{dt} = \frac{dC^*}{dt}(B^*C^*)^{-1} - C^*(B^*C^*)^{-1}\left[\frac{dB^*}{dt}C^* + B^*\frac{dC^*}{dt}\right](B^*C^*)^{-1},$$

we obtain

$$\lambda^d_T|_{t=T} = -\dot{\lambda}^d - C^*(B^*C^*)^{-1}\left[\frac{dg^*_{x^a}}{dt} - \frac{dB^*}{dt}C^*(B^*C^*)^{-1}g^*_{x^a}\right] - P^*\frac{dC^*}{dt}(B^*C^*)^{-1}g^*_{x^a}.$$

But we know that at $t = T$, $\lambda^d = -C^*(B^*C^*)^{-1}g^*_{x^a}$. Thus

$$\lambda^d_T|_{t=T} = -\dot{\lambda}^d - C^*(B^*C^*)^{-1}\left[\frac{dg^*_{x^a}}{dt} + \frac{dB^*}{dt}\lambda^d\right] - P^*\frac{dC^*}{dt}(B^*C^*)^{-1}g_{x^a}.$$

Taking the derivative of $B^*\lambda^d + g^*_{x^a} = 0$, we have

$$B^*\dot{\lambda}^d = -\left[\frac{dg^*_{x^a}}{dt} + \frac{dB^*}{dt}\lambda^d\right].$$

Substituting for $\dot{\lambda}^d$, finally we obtain the boundary condition for λ^d_T:

$$\lambda^d_T(T,T) = P^*\left[g^*_{x^d} + A^*\lambda^d(T,T) - \frac{dC^*}{dt}(B^*C^*)^{-1}g^*_{x^a}\right].$$

4. Stability. In this section we investigate the stability of the adjoint system. That is, if the original system is stable, will the adjoint system also be stable? If we just consider the adjoint equation (7), it may not be stable. Consider the following example:

$$e^t \dot{x} + \frac{1}{2} e^t x = 0. \tag{22}$$

This system is equivalent to

$$\dot{x} + \frac{1}{2} x = 0,$$

so it is stable. But the adjoint system (7) for (22) is

$$e^t \dot{\lambda} - \frac{1}{2} e^t \lambda + e^t \lambda = 0, \tag{23}$$

which is equivalent to

$$\dot{\lambda} + \frac{1}{2} \lambda = 0. \tag{24}$$

Note that we solve the adjoint system in the backward direction. Thus the adjoint system (24) is unstable.

Denoting $\bar{\lambda} = F_{\dot{x}}^* \lambda$, we can form the *augmented adjoint system* for (7):

$$\begin{cases} \dot{\bar{\lambda}} - F_x^* \lambda = -g_x^*, \\ \bar{\lambda} - F_{\dot{x}}^* \lambda = 0. \end{cases} \tag{25}$$

If we solve the augmented adjoint system (25) instead of (24), $\bar{\lambda}$ satisfies

$$\dot{\bar{\lambda}} - \frac{1}{2} \bar{\lambda} = 0, \tag{26}$$

which is stable in the backward direction. We will show that in general the augmented adjoint system (25) for $\bar{\lambda}$ is stable if the original system is stable. Thus in the implementation [12] we solve (25) instead of (7).

We show first that the adjoint system (7) for explicit ODE, semi-explicit index-1 DAE, and Hessenberg index-2 DAE systems preserves the stability. In these cases, there is no difference between λ and $\bar{\lambda}$. Then by bounded transformation, we show that the stability of the augmented adjoint system (25) for $\bar{\lambda}$ is also preserved for implicit ODE and general index-1 DAEs.

4.1. Explicit ODE. For a linear ODE $\dot{x} = A(t)x$, the corresponding homogeneous adjoint system is

$$\dot{\lambda} = -A^* \lambda.$$

Since the adjoint system is solved backward, we can do a change of variable $\tau = T - t$. The adjoint system, now to be solved forward, is transformed to

$$\dot{\lambda} = A^* \lambda.$$

Then we have the following well-known result.

THEOREM 4.1. *If the ODE system $\dot{x} = A(t)x$ is stable, the adjoint ODE system is also stable.*

Proof. If A is constant, the result follows from the fact that A and A^* have the same eigenvalues. If A is not constant, we look at the Green's function. Let X and Λ be the fundamental solutions of the original ODE and the adjoint system, respectively. The original system is stable if and only if $\|X(s)X^{-1}(t)\|$ is bounded for $s > t$. The adjoint system is stable if and only if $\|\Lambda(s)\Lambda^{-1}(t)\|$ is bounded for $s > t$.

Let $Z(t) = X^*(T-t)\Lambda(t)$. Then

$$\dot{Z} = -\dot{X}^*(T-t)\Lambda(t) + X^*(T-t)\dot{\Lambda}(t) = -X^*A^*\Lambda + X^*A^*\Lambda = 0.$$

Thus $Z(t)$ is constant, so that

$$X^*(T-t)\Lambda(t) = X^*(T-s)\Lambda(s).$$

Then

$$\Lambda(s)\Lambda^{-1}(t) = \left[X(T-t)X^{-1}(T-s)\right]^*.$$

Finally, note that $s > t$ leads to $T - t > T - s$. ◻

4.2. Semi-explicit index-1 DAE. Consider the linear semi-explicit index-1 DAE system

$$\begin{cases} \dot{x}^d = A(t)x^d + B(t)x^a, \\ 0 = C(t)x^d + D(t)x^a, \end{cases}$$

where $D(t)$ is invertible. The corresponding adjoint system is

$$\begin{cases} \dot{\lambda}^d = -A^*\lambda^d - C^*\lambda^a, \\ 0 = B^*\lambda^d + D^*\lambda^a. \end{cases}$$

The original system is stable if and only if the essential underlying ODE (EUODE) $\dot{x}^d = Ax^d - B(D)^{-1}Cx^d$ is stable [1]. The corresponding EUODE for the adjoint system is given by $\dot{\lambda}^d = -A^*\lambda^d + C^*(D^*)^{-1}B^*\lambda^d$, or $\dot{\lambda}^d = -(A - B(D)^{-1}C)^*\lambda^d$. Thus the EUODE of the adjoint system is the adjoint of the EUODE of the original system. We know from Theorem 4.1 that the EUODE of the adjoint system is stable. Thus the adjoint system is also stable.

4.3. Hessenberg index-2 DAE. Consider the linear Hessenberg index-2 DAE system

$$\begin{cases} \dot{x}^d = A(t)x^d + B(t)x^a + q(t), \\ 0 = C(t)x^d + r(t), \end{cases}$$

where $C(t)B(t)$ is nonsingular. The corresponding EUODE is derived in [1] as follows. If B is sufficiently smooth, there exists a smooth bounded matrix function $R(t) \in R^{(n_d - n_a)\times n_d}$ whose linearly independent, normalized rows form a basis for the nullspace of B^*. Here n_d is the dimension of the differential variables x^d and n_a is the dimension of the algebraic variables x_a. Thus $R(t)B(t) = 0$ and

$$\begin{pmatrix} R \\ C \end{pmatrix} \text{ is invertible.}$$

Defining new variables

$$u = Rx^d, \quad 0 \le t \le T, \tag{27}$$

x^d is given by

$$x^d = \begin{pmatrix} R \\ C \end{pmatrix}^{-1} \begin{pmatrix} u \\ -r \end{pmatrix} = Su - Fr, \tag{28}$$

where $S(t) \in R^{n_d \times (n_d - n_a)}$ satisfies

$$RS = I, \quad CS = 0. \tag{29}$$

The corresponding EUODE is

$$\dot{u} = (RAS + \dot{R}S)u - (RAF - \dot{R}F)r(t) + Rq(t). \tag{30}$$

The original system is stable for the differential variables if the EUODE (30) is stable.

Now consider the adjoint system

$$\begin{cases} \dot{\lambda}^d = -A^*(t)\lambda^d - C^*(t)\lambda^a + \hat{q}(t), \\ 0 = B^*(t)\lambda^d + \hat{r}(t). \end{cases}$$

Since $CS = 0$, we have $S^*C^* = 0$. Multiplying the adjoint system by $S^*(t)$, we obtain

$$S^*\lambda = -S^*A^*(t)\lambda + S^*\hat{q}(t). \tag{31}$$

Letting $v = S^*\lambda$, we have

$$\lambda^d = \begin{pmatrix} S^* \\ B^* \end{pmatrix} \begin{pmatrix} v \\ -\hat{r} \end{pmatrix} = \begin{pmatrix} R^* & C^*(B^*C^*)^{-1} \end{pmatrix} \begin{pmatrix} v \\ -\hat{r} \end{pmatrix}. \tag{32}$$

Thus the EUODE of the adjoint DAE is

$$\dot{v} = \dot{S}^*R^*v - \dot{S}^*C^*(B^*C^*)^{-1}r - S^*A^*R^*v + S^*A^*C^*(B^*C^*)^{-1}r. \tag{33}$$

$RS = I$ leads to $\dot{S}^*R^* = -S^*\dot{R}^*$, so the homogeneous part of the EUODE is given by

$$\dot{v} = -(S^*\dot{R}^* + S^*A^*R^*)v = -(\dot{R}S + RAS)^*v. \tag{34}$$

We can see that the EUODE of the adjoint system is the same as the adjoint of the EUODE of the original system. Thus from Theorem 4.1 we know that the adjoint system for a stable linear Hessenberg index-2 DAE is stable for the differential variables.

4.4. Fully implicit DAE. For implicit ODE and index-1 DAEs, we have already seen from the example (24) that the adjoint DAE (7) for λ may be unstable. But we will show that the augmented adjoint system (25) for $\bar{\lambda}$ preserves the stability. To prove that, we first refer to a lemma from Campbell, Bichols, and Terrel [11] that adjoint commutes with any nonsingular linear transformation.

LEMMA 4.2. *Given the time-dependent linear DAE system*

$$A(t)\dot{x} + B(t)x = f(t) \tag{35}$$

and nonsingular time-dependent differentiable matrices $P(t)$ multiplying the equations of the DAE and $Q(t)$ transforming the variables, the adjoint system of the transformed DAE is the transformed system of the adjoint DAE.

Lemma 4.2 is valid for any P and Q. When we consider stability, we require that the transformation matrix be bounded for all t. First, the matrix Q should be bounded to ensure that the variable change $x = Q(t)\tilde{x}$ will not alter the stability. Second, the matrix P should be bounded so that the variable change $\lambda = P^*\tilde{\lambda}$ will not change the stability. But from the example (22), we can see that in the case that $A(t)$ is unbounded but the solution is still bounded, P and Q cannot both be bounded. Thus we require that Q be bounded. Instead of requiring that P be bounded, we require that PA be bounded. Then the variable change $\bar{\lambda} = A^*P^*\tilde{\lambda}$ will not alter the stability. Thus, the augmented adjoint system (25) preserves the stability.

To illustrate this point, consider again the example (22). The transformation matrix P is e^{-t}, which is unbounded in the backward direction. If we set $\lambda = P^*\tilde{\lambda}$ as in Theorem 4.3, λ will not be bounded. But $A = e^t$, so PA is the identity. If we let $\bar{\lambda} = A^*\lambda = A^*P^*\tilde{\lambda}$, $\bar{\lambda}$ will have the same stability as $\tilde{\lambda}$. In general, the transformation in the adjoint (7) may not be bounded but the transformation in the augmented adjoint system (25) for $\bar{\lambda}$ will be. Thus we have the following theorem.

THEOREM 4.3. *For general index-0 and index-1 DAE systems, if the original DAE system is stable, the augmented adjoint DAE system (25) for $\bar{\lambda}$ is stable.*

Proof. Using the smooth SVD [13] of $A(t)$, we can construct orthogonal matrices Q and $\tilde{P}$ so that

$$\tilde{P}(t)A(t)Q(t) = \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix}, \tag{36}$$

where $\Sigma = \text{diag}\{\lambda_1, \ldots, \lambda_k\}$, $\lambda_i \neq 0$. Defining $P(t) = \begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & I \end{bmatrix} \tilde{P}(t)$, we have

$$P(t)A(t)Q(t) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}. \tag{37}$$

Let $x = Q\tilde{x}$. Since Q is orthogonal, if the original system is stable, the transformed DAE system is also stable. From the conclusion for the explicit cases, we know that the adjoint of the transformed DAE system is also stable. Now, instead of letting $\lambda = P^*\tilde{\lambda}$, we let $\bar{\lambda} = A^*\lambda = (PA)^*\tilde{\lambda}$. Since

$$PA = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} Q^*,$$

it follows that PA is bounded. Thus the augmented adjoint system (25) for $\bar{\lambda}$ is stable. ∎

Remark. From the above proof, we can see that if P is bounded, λ is bounded. P will be bounded if the pseudoinverse of A is bounded.

5. Numerical stability. In this section we consider numerical stability for the adjoint system. For systems of index-0 or index-1, we consider the general linear DAE

$$A(t)\dot{x}(t) + B(t)x(t) = 0. \tag{38}$$

The corresponding adjoint system, which is solved in the reverse direction, is

$$(A^*\lambda)' - B^*\lambda = 0. \tag{39}$$

Zero-stability, the stability as the stepsize $h \to 0$ and the number of steps $n \to \infty$ on a fixed time interval, has been thoroughly investigated for ODEs, index-1, and Hessenberg index-2 DAEs. See [18, 1, 16, 17], where it is shown that numerical ODE methods such as the backward Euler method when applied to well-posed DAEs of those classes are zero-stable and convergent. Zero-stability for the adjoint system follows immediately for ODEs. For index-1 DAEs, zero-stability is a consequence of the index-1 structure of the adjoint system [7]. It is simple to verify that the adjoint system for an index-2 Hessenberg DAE is index-2 Hessenberg, from which zero-stability follows immediately.

Asymptotic numerical stability, the stability for a fixed stepsize h and $n \to \infty$, has been thoroughly studied for ODE and DAE systems [18, 2, 1, 16, 17]. Here we are concerned with the question, If a numerical method with a given stepsize is stable for the original system (38), will it also be stable for the adjoint system (39)? We consider asymptotic numerical stability for the backward Euler method.

Discretizing the original system (38), we obtain

$$A_{n+1}\left(\frac{x_{n+1}-x_n}{h}\right) + B_{n+1}x_{n+1} = 0, \tag{40}$$

which leads to

$$x_{n+1} = \left[A_{n+1} + hB_{n+1}\right]^{-1} A_{n+1}x_n. \tag{41}$$

For the adjoint system (39), we have (solving backward)

$$\left(\frac{A^*_{n+1}\lambda_{n+1} - A^*_n\lambda_n}{h}\right) - B^*_n\lambda_n = 0, \tag{42}$$

which leads to

$$\lambda_n = \left[A^*_n + hB^*_n\right]^{-1} A^*_{n+1}\lambda_{n+1}. \tag{43}$$

Thus we have

$$\begin{aligned}\lambda_0 &= \left[(A^*_0 + hB^*_0)^{-1} A^*_1\right] \cdots \left[(A^*_n + hB^*_n)^{-1} A^*_{n+1}\right] \cdots \left[\left(A^*_{N-1} + hB^*_{N-1}\right)^{-1} A^*_N\right] \lambda_N \\ &= (A^*_0 + hB^*_0)^{-1} \cdots \left[A^*_n \left(A^*_n + hB^*_n\right)^{-1}\right] \cdots \left[A^*_{N-1}\left(A^*_{N-1} + hB^*_{N-1}\right)^{-1}\right] A^*_N\lambda_N.\end{aligned}$$

Taking the transpose, we obtain

$$\lambda^*_0 = \lambda^*_N A_N (A_{N-1} + hB_{N-1})^{-1} A_{N-1} \cdots (A_1 + hB_1)^{-1} A_1 (A_0 + hB_0)^{-1}. \tag{44}$$

Let

$$C_N = (A_{N-1} + hB_{N-1})^{-1} A_{N-1} \cdots (A_1 + hB_1)^{-1} A_1.$$

The numerical stability of the original system implies that C_N is bounded. From (44) we have for the adjoint system

$$\lambda^*_0 = \lambda^*_N A_N C_N (A_0 + hB_0)^{-1}. \tag{45}$$

Thus it follows for all linear ODEs and DAEs that asymptotic numerical stability of the backward Euler method for the original system implies asymptotic numerical stability for the adjoint system.

Note that the backward Euler discretization (42) which has been assumed in the stability analysis is not the same as the discretization which would normally be used in software like DASPK3.0 for solving the adjoint system (39). The difference occurs for systems that are fully implicit. In order to solve (39) using DASPK3.0, we would need to first rewrite (39) to isolate the time derivative $\dot{\lambda}$:

$$A^*\dot{\lambda} + A^{*\prime}\lambda - B^*\lambda = 0. \tag{46}$$

Discretization by the backward Euler method would then yield (solving backward)

$$A_n^*\left(\frac{\lambda_{n+1}-\lambda_n}{h}\right) + A_n^{*\prime}\lambda_n - B_n^*\lambda_n = 0, \tag{47}$$

which differs from the discretization (42) and *in general does not preserve the asymptotic numerical stability of the forward system.*

Consider example (22), whose backward Euler discretization leads to

$$x_{n+1} = \left(1 + \frac{1}{2}h\right)^{-1} x_n, \tag{48}$$

which is stable for any $h > 0$. The adjoint system is

$$e^t\dot{\lambda} - \frac{1}{2}e^t\lambda + e^t\lambda = 0. \tag{49}$$

The corresponding backward Euler discretization is given by

$$e^{t_n}\left(\frac{\lambda_{n+1}-\lambda_n}{h}\right) + e^{t_n}\lambda_n - \frac{1}{2}e^{t_n}\lambda_n = 0, \tag{50}$$

which is equivalent to

$$\lambda_n = \left(1 - \frac{1}{2}h\right)^{-1}\lambda_{n+1}, \tag{51}$$

which is unstable for any $0 < h < 4$. Thus this discretization is not stable using stepsizes for which the discretization of the forward problem is stable.

The discretization (42), which as we have seen preserves the numerical stability, can be achieved in DAE software by solving the *augmented adjoint system* (25).

Acknowledgments. The authors would like to thank Steve Lee, Peter Brown, Keith Grant, Luc Machiels, and Alan Hindmarsh of Lawrence Livermore National Laboratory for bringing this problem to our attention and for several interesting discussions.

REFERENCES

[1] U. M. Ascher and L. R. Petzold, *Stability of computational methods for constrained dynamics systems*, SIAM J. Sci. Comput., 14 (1993), pp. 95–120.

[2] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, Philadelphia, 1998.

[3] K. Balla, *Linear subspaces for linear DAEs of index* 1, Comput. Math. Appl., 32 (1996), pp. 81–86.

[4] K. Balla, *Boundary conditions and their transfer for differential algebraic equations of index* 1, Comput. Math. Appl., 31 (1996), pp. 1–5.

[5] K. Balla and R. März, *Transfer of boundary conditions for DAEs of index* 1, SIAM J. Numer. Anal., 33 (1996), pp. 2318–2332.

[6] K. Balla and R. März, *An Unified Approach to Linear Differential Algebraic Equations and Their Adjoint Equations*, Institute of Mathematics Technical report, Humboldt University, Berlin, 2000.

[7] K. Balla and R. März, *Linear differential algebraic equations of index* 1 *and their adjoint equations*, Results in Mathematics, 37 (2000), pp. 13–35.

[8] C. Bischof, A. Carle, G. Corliss, A. Griewank, and P. Hovland, *ADIFOR - Generating derivative codes from Fortran programs*, Scientific Programming, 1 (1992), pp. 11–29.

[9] K. E. Brenan, S. L. Campbell, and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, SIAM, Philadelphia, 1995.

[10] P. N. Brown, A. C. Hindmarsh, and L. R. Petzold, *Using Krylov methods in the solution of large-scale differential-algebraic systems*, SIAM J. Sci. Comput., 15 (1994), pp. 1467–1488.

[11] S. L. Campbell, N. K. Bichols, and W. J. Terrel, *Duality, observability and controllability for linear time-varying systems*, Circuits, Systems, Signal Process, 10 (1991), pp. 455–470.

[12] Y. Cao, S. Li, and L. R. Petzold, *Adjoint sensitivity analysis for differential-algebraic equations: Algorithms and software*, J. Comput. Appl. Math., to appear.

[13] L. Dieci and T. Eirola, *On smooth decompositions of matrices*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 800–819.

[14] R. M. Errico, *What is an adjoint model?*, Bulletin of the American Meteorological Society, 78 (1997), pp. 2577–2591.

[15] W. F. Feehery, J. E. Tolsma, and P. I. Barton, *Efficient sensitivity analysis of large-scale differential-algebraic systems*, Appl. Numer. Math., 25 (1997), pp. 41–54.

[16] E. Hairer, S. P. Norsett, and G. Wanner, *Solving Ordinary Differential Equations* I, Springer–Verlag, Berlin, 1987.

[17] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations* II, *Stiff and Differential-Algebraic Problems*, Springer–Verlag, Berlin, 1991.

[18] M. Hanke, E. I. Macana, and R. März, *On asymptotics in case of linear index-*2 *differential-algebraic equations*, SIAM J. Numer. Anal., 35 (1998), pp. 1326–1346.

[19] D. Knapp, V. Barocas, K. Yoo, L. R. Petzold, and R. Tranquillo, *Rheology of reconstituted type* I *collagen gel in confined compression*, J. Rheology, 41 (1997), pp. 971–993.

[20] S. Li and L. R. Petzold, *Software and algorithms for sensitivity analysis of large-scale differential-algebraic systems*, J. Comput. Appl. Math., 125 (2000), pp. 131–145.

[21] S. Li and L. R. Petzold, *Design of New DASPK for Sensitivity Analysis*, Technical report, Dept. of Computer Science, UCSB, Santa Barbara, CA, 1999.

[22] S. Li, L. R. Petzold, and W. Zhu, *Sensitivity analysis of differential-algebraic equations: A comparison of methods on a special problem*, Appl. Numer. Math., 32 (2000), pp. 161–174.

[23] T. Maly and L. R. Petzold, *Numerical methods and software for sensitivity analysis of differential-algebraic systems*, Appl. Numer. Math., 20 (1997), pp. 57–79.

[24] R. März, *Differential Algebraic Systems Anew*, Institute of Mathematics Technical Report, Humboldt University, Berlin, 2000.

[25] L. L. Raja, R. J. Kee, R. Serban, and L. R. Petzold, *Computational algorithm for dynamic optimization of chemical vapor deposition processes in stagnation flow reactors*, J. Electrochemical Soc., 147 (2000), pp. 2718–2726.