Computer Science Technical Report
TR-07-40
November 27, 2007

Adrian Sandu

"On Consistency Properties of Discrete Adjoint Linear Multistep Methods"

Computer Science Department
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061
Phone: (540)-231-2193
Fax: (540)-231-9218
Email: sandu@cs.vt.edu
Web: http://www.eprints.cs.vt.edu

VirginiaTech
Invent the Future

ON CONSISTENCY PROPERTIES
OF DISCRETE ADJOINT LINEAR MULTISTEP METHODS

ADRIAN SANDU
DEPARTMENT OF COMPUTER SCIENCE
VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
BLACKSBURG, VA 24061
TEL: 540-231-2193, FAX: 540-231-9218, EMAIL: (SANDU@CS.VT.EDU).

Abstract. In this paper we analyze the consistency properties of discrete adjoints of linear multistep methods. Discrete adjoints are very popular in optimization and control since they can be constructed automatically by reverse mode automatic differentiation. The consistency analysis reveals that the discrete linear multistep adjoints are, in general, inconsistent approximations of the adjoint ODE solution along the trajectory. However, the discrete adjoints at the initial time (and therefore the discrete adjoint gradients) converge to the adjoint ODE solution with the same order as the original linear multistep method. Discrete adjoints inherit the zero-stability properties of the forward method. Numerical results confirm the theoretical findings.

Keywords: Linear multistep methods, discrete adjoints.

1. Introduction.

Consider an ordinary differential equation (ODE) whose evolution depends on a vector of parameters

$$x' = \phi(t,x,\theta)\ , \quad x\left(t_{\rm ini}\right) = x_{\rm ini}(\theta)\ , \quad t_{\rm ini} \le t \le t_{\rm end}\ . \tag{1.1}$$

Here $x \in \Re^{n_x}$ and $\theta \in \Re^{n_p}$. Consider also the following general cost function whose value depends on the solution of the ODE:

$$\overline{\Psi}\left(\theta\right) = \int_{t_{\rm ini}}^{t_{\rm end}} u\big(t,x(t,\theta),\theta\big)\,dt + v\big(t_{\rm end},x(t_{\rm end},\theta),\theta\big) \quad \text{subject to (1.6)}\ . \tag{1.2}$$

We are interested to find the parameter values for which the cost function is minimized,

$$\min_{\theta} \overline{\Psi}\left(\theta\right) \quad \text{subject to (1.1)}\ . \tag{1.3}$$

The general optimization problem (1.3) arises in many important applications including control, shape optimization, parameter identification, data assimilation, etc. To apply a gradient based numerical optimization procedure one needs to compute the gradient of the cost function $\overline{\Psi}$ with respect to the parameters

$$\nabla_\theta \overline{\Psi} = \left(\frac{\partial \overline{\Psi}}{\partial \theta}\right)^T\ . \tag{1.4}$$

Here and throughout the paper the derivatives of scalar functions with respect to vectors of arguments are row vectors (e.g., $\partial\overline{\Psi}/\partial\theta$ is an n_p-dimensional row vector).

The optimization problem (1.1)–(1.2)–(1.3) can be reformulated as follows. Extend formally the ODE (1.1) with dummy equations for time and the parameters, and with a "quadrature equation" for the integral term in (1.2):

$$\left[\begin{array}{c} t \\ x \\ \theta \\ z \end{array}\right]' = \left[\begin{array}{c} 1 \\ \phi(t,x,\theta) \\ 0 \\ u\big(t,x,\theta\big) \end{array}\right]\ , \quad \left[\begin{array}{c} t \\ x \\ \theta \\ z \end{array}\right]\big(t_{\rm ini}\big) = \left[\begin{array}{c} t_{\rm ini} \\ x_{\rm ini}(\theta) \\ \theta \\ 0 \end{array}\right]\ , \quad t_{\rm ini} \le t \le t_{\rm end}\ . \tag{1.5}$$

The extended state vector in (1.5) has a dimension $d = 1 + n_x + n_p + 1$. With the compact notation

$$y = \begin{bmatrix} t \\ x \\ \theta \\ z \end{bmatrix} \in \Re^d \ , \quad f(y) = \begin{bmatrix} 1 \\ \phi(y) \\ 0 \\ u(y) \end{bmatrix} \ , \quad g(y) = g(t,x,\theta,z) = z + v\big(t,x,\theta\big) \ ,$$

the differential equation (1.1) is equivalent to the following ODE

$$y' = f(y) \ , \quad y\left(t_{\rm ini}\right) = y_{\rm ini} \ , \quad t_{\rm ini} \le t \le t_{\rm end} \ . \tag{1.6}$$

Using the quadrature variables the cost function (1.2) can be expressed in terms of the solution of the extended system (1.5) at the final time:

$$\overline{\Psi}(\theta) = \overline{\Psi}(y_{\rm ini}) = z(t_{\rm end}) + v\big(t_{\rm end}, x(t_{\rm end},\theta),\theta\big) = g\big(y(t_{\rm end})\big) \ . \tag{1.7}$$

This cost function depends on the initial conditions of (1.6), which contain the original parameters θ.

Without loss of generality the optimization problem (1.3) can be posed as follows:

$$\min_{y_{\rm ini}} \overline{\Psi}\left(y_{\rm ini}\right) = g\big(y(t_{\rm end})\big) \quad \text{subject to (1.6)} \ . \tag{1.8}$$

Throughout the paper we assume that the (extended) ODE function (1.8) is sufficiently smooth, i.e., it has sufficiently many continuous derivatives as required to have the numerical solutions converge at the appropriate orders. The Jacobian of the (extended) ODE function (1.6) is denoted by $J(y) = \partial f(y)/\partial y \in \Re^{d\times d}$. We make the additional assumption that for all cases of interest the initial condition and the entire smooth solution of (1.8) remain within a bounded set Ω (which can be large),

$$y_{\rm ini} \in \Omega \quad \Rightarrow \quad y(t) \in \Omega \ \ \forall\, t \in [t_{\rm ini}, t_{\rm end}] \ .$$

Consequently throughout the trajectory the norms of the ODE function and its Jacobian remain bounded throughout all trajectories of interest

$$\|f(y)\| \le C \ , \quad \|J(y)\| \le M \ , \quad \forall\, y \in \Omega \ .$$

To solve (1.6)–(1.8) via a gradient based optimization procedure one needs to compute the derivatives of the cost function $\overline{\Psi}$ with respect to the initial conditions. This can be done effectively using continuous or discrete adjoint modeling. These approaches are discussed next.

In the *continuous adjoint* ("differentiate-then-discretize") approach [8] one derives the adjoint ODE associated with (1.6)

$$\overline{\lambda}' = -J^T\Big(t, y(t)\Big)\,\overline{\lambda} \ , \quad \overline{\lambda}\left(t_{\rm end}\right) = \left(\frac{\partial g}{\partial y}\big(y(t_{\rm end})\big)\right)^T \ , \quad t_{\rm end} \ge t \ge t_{\rm ini} \ , \tag{1.9}$$

The system (1.9) is solved backwards in time from $t_{\rm end}$ to $t_{\rm ini}$ to obtain the gradients of the cost function with respect to the state [8]

$$\overline{\lambda}(t) = \left(\frac{\partial \overline{\Psi}}{\partial y(t)}\right)^T \ , \quad \overline{\lambda}(t_0) = \left(\frac{\partial \overline{\Psi}}{\partial y_{\rm ini}}\right)^T \ . \tag{1.10}$$

Note that the continuous adjoint equation (1.9) depends on the forward solution $y(t)$ via the argument of the Jacobian.

For a computer implementation the continuous adjoint ODE (1.9) is discretized using a numerical integration technique. Numerical solutions $\overline{\lambda}_n \approx \overline{\lambda}(t_n)$ are obtained at the discrete time moments $t_{\rm end} = t_N > t_{N-1} > \cdots > t_1 > t_0 = t_{\rm ini}$. These solutions are numerical approximations of the continuous adjoint sensitivities

$$\overline{\lambda}_n \approx \left(\frac{\partial \overline{\Psi}}{\partial y(t_n)} \right)^T \quad , \quad \overline{\lambda}_0 \approx \left(\frac{\partial \overline{\Psi}}{\partial y_{\rm ini}} \right)^T \quad . \tag{1.11}$$

In the *discrete adjoint* ("discretize-then-differentiate") approach [8] one starts with a numerical discretization of the forward ODE (1.6). Numerical approximations of the ODE solution $y_n \approx y(t_n)$ are available the discrete time moments $t_{\rm ini} = t_0 < t_1 < \cdots < t_N = t_{\rm end}$

$$y_0 = y_{\rm ini} \ , \quad y_n = \mathcal{M}_n \left(y_0, \cdots, y_{n-1} \right) \ , \quad n = 1, \cdots, N \ . \tag{1.12}$$

The numerical solution at the final time is $y_N \approx y\left(t_{\rm end}\right)$. The optimization problem (1.8) is reformulated in terms of the numerical solution minimized,

$$\min_{y_{\rm ini}} \Psi \left(y_{\rm ini} \right) = g\big(y_N\big) \quad \text{subject to (1.12)} \ . \tag{1.13}$$

The gradient of (1.13) is computed directly from (1.12) using the transposed chain rule. This calculation proceeds backwards in time, and produces the discrete adjoint variables λ_N, λ_{N-1}, $\cdots$, λ_0

$$\begin{aligned} \lambda_N &= \left(\frac{\partial g}{\partial y}(y_N) \right)^T \quad , \quad \lambda_n = 0 \ , \quad n = N-1, \cdots, 0 \ , \qquad\qquad (1.14)\\ \lambda_\ell &= \lambda_\ell + \left(\frac{\partial \mathcal{M}_n}{\partial y_\ell}(y_0, \cdots, y_{n-1}) \right)^T \lambda_n \ , \quad \ell = n-1, \cdots, 0 \ , \quad n = N, \cdots, 0 \ . \end{aligned}$$

Note that the discrete adjoint equation (1.14) depends on the forward numerical solution $y_0, \cdots, y_N$ via the arguments of the discrete model. The discrete adjoint process gives the sensitivities of the numerical cost function (1.13) with respect to changes in the forward numerical solution (1.12)

$$\lambda_n = \left(\frac{\partial \Psi}{\partial y_n} \right)^T \quad , \quad \lambda_0 = \left(\frac{\partial \Psi}{\partial y_{\rm ini}} \right)^T \quad . \tag{1.15}$$

Continuous adjoints are useful for sensitivity analysis studies, as well as for optimization. They are computed by applying the numerical solver of choice to the continuous equation (1.9), and using the forward solution $y(t)$ obtained by interpolation from a sequence of checkpoints. Discrete adjoints are useful in optimization since they provide the gradients of the numerical function that is being numerically minimized (1.13). The discrete adjoint process (1.14) can be obtained by applying automatic differentiation in reverse mode to the (existing) implementation of the forward model (1.12).

The focus of this paper is to analyze the relationship between the discrete adjoint variables (1.15) and the continuous adjoint solution (1.10) when the underlying ODE model (1.9) is solved using linear multistep methods (1.12). The choice of the forward discretization method (1.12) fully determines the properties of the discrete adjoint process (1.14). We regard the discrete adjoint process (1.14) as a numerical method applied to solve the adjoint ODE (1.9) and study its consistency properties.

Consistency properties of discrete Runge-Kutta adjoints have been studied by Hager [3], who gives additional order conditions necessary in the context of control problems. Walther [10] has studied the effects of reverse mode automatic differentiation on explicit Runge-Kutta methods in control, and finds that the order of the discretization is preserved by discrete adjoints. Giles [2] has discussed Runge-Kutta adjoints in the context of steady state flows. In this paper we consider control problems where only the initial conditions are the control variables. This setting is simpler than the distributed control case considered in [3, 10].

Sandu et al. have shown the consistency of discrete Runge Kutta adjoints with the adjoint ODE solution [7]. Efficient implementation aspects for discrete Runge Kutta adjoints have been discussed in [6]. Second order Runge Kutta adjoints and their properties have been studied studied in ([9]).

Baguer et al. [1] have constructed discrete adjoints for linear multistep methods in the context of control problems. Their work does not discuss the consistency of these adjoints with the adjoint ODE solution, therefore the quality of these gradients.

In this paper we discuss the consistency of discrete adjoints of linear multistep methods (LMM) with the adjoint ODE. The analysis is carried out under the following conditions. The cost function depends (only) and the final solution values, and the (only) control variables are the initial conditions. The system of ODEs and its solution are continuously differentiable sufficiently many times to make the discussion of order of consistency meaningful. The analysis assumes small time steps, such that the error estimates hold for for non-stiff systems. The sequence of (possibly variable) step sizes in the forward integration is predefined. The possible dependency of the step size on the forward solution is not considered in the adjoint calculations. Such a dependency can be introduced by a step size control mechanism based on error estimates. The assumption of predefined steps means that the step control mechanism is not differentiated. When using automatic differentiation special directives may be used to instruct the parser to not differentiate the controller part of the code.

The paper is organized as follows. In Section 2 we derive the discrete adjoints for linear multistep methods. The consistency of the discrete adjoints with the adjoint ODE *along the entire trajectory* is studied in Section 2.1 for fixed step size integration and in Section 2.2 for variable step integration. Zero-stability properties are discussed in Section 3. The consistency of the discrete adjoint *at the initial time* is analyzed in Section 4. The numerical results in Section 5 confirm the theoretical findings. Conclusions are drawn in Section 6.

2. Linear Multistep Methods.

Consider the linear multistep method

$$y_0 = y_{\rm ini} \;, \tag{2.1a}$$

$$y_n = \theta_n\left(y_0, \cdots, y_{n-1}\right) \;, \quad n = 1, \cdots, k-1 \;, \tag{2.1b}$$

$$\sum_{i=0}^{k} \alpha_i^{[n]}\, y_{n-i} = h_n \sum_{i=0}^{k} \beta_i^{[n]}\, f_{n-i} \;, \quad n = k, \cdots, N \;. \tag{2.1c}$$

The upper indices indicate the dependency of the method coefficients on the step number; this formulation accommodates variable step sizes. The numerical solution is computed at the discrete moments $t_{\rm ini} = t_0 < t_1 < \cdots < t_N = t_{\rm end}$. As usual y_n represents the numerical approximation at time t_n. The right hand side function evaluated at t_n using the numerical solution y_n is denoted $f_n = f(t_n, y_n)$, while its Jacobian is denoted by $J_n = J(t_n, y_n) = (\partial f / \partial y)\,(t_n, y_n)$.

The discretization time steps and their ratios are

$$h_n = t_n - t_{n-1}\ , \quad n = 1, \cdots, N\ ; \qquad \omega_n = \frac{h_n}{h_{n-1}}\ , \quad n = 2, \cdots, N\ . \tag{2.2}$$

We denote the sequence of discretization step sizes and the maximum step size by

$$h = \big(h_1, \cdots, h_N\big) \quad \text{and} \quad |h| = \max_{1 \le n \le N} h_n\ . \tag{2.3}$$

The number of steps depends on the step discretization sequence, $N = N(h)$.

Equation (2.1a)–(2.1c) is a k-step method. The method coefficients $\alpha_i^{[n]}$, $\beta_i^{[n]}$ depend on the sequence of (possibly variable) steps, specifically, they depend on the ratios $\omega_{n-k+2}, \cdots, \omega_n$.

A starting procedure θ is used to produce approximations of the solution $y_i = \theta_i\,(y_0, \cdots, y_{i-1})$ at times t_i, $i = 1, \cdots, k-1$. We will consider the starting procedures to be linear numerical methods. This setting covers both the case of self-starting LMM methods (a linear i-step method gives y_i for $i = 1, \cdots, k-1$) as well as the case where a Runge Kutta method is used for initialization ($y_i = \theta_i\,(y_{i-1})$ for $i = 1, \cdots, k-1$).

We next derive the discrete adjoint method associated with (2.1a)–(2.1c). The following result is a generalization of [1, Corollary 3.4].

Proposition 2.1 (The discrete LMM adjoint process).

The discrete adjoint method associated with the linear multistep method (2.1a)–(2.1c) and the cost function

$$\Psi\,(y_{\rm ini}) = g\big(y_N\big)$$

reads:

$$\alpha_0^{[N]} \lambda_N = h_N\, \beta_0^{[N]}\, J_N^T \cdot \lambda_N + \left(\frac{\partial g}{\partial y}\,(y_N)\right)^T\ , \tag{2.4a}$$

$$\sum_{i=0}^{N-m} \alpha_i^{[m+i]}\, \lambda_{m+i} = J_m^T \cdot \sum_{i=0}^{N-m} h_{m+i}\, \beta_i^{[m+i]}\, \lambda_{m+i}\ , \tag{2.4b}$$
$$m = N-1, \cdots, N-k+1\ ,$$

$$\sum_{i=0}^{k} \alpha_i^{[m+i]}\, \lambda_{m+i} = J_m^T \cdot \sum_{i=0}^{k} h_{m+i}\, \beta_i^{[m+i]}\, \lambda_{m+i}\ , \tag{2.4c}$$
$$m = N-k, \cdots, k\ ,$$

$$\lambda_{k-1} + \sum_{i=1}^{k} \alpha_i^{[k-1+i]}\, \lambda_{k-1+i} = J_{k-1}^T \cdot \sum_{i=1}^{k} \Big(h_{k-1+i}\, \beta_i^{[k-1+i]}\, \lambda_{k-1+i}\Big) \tag{2.4d}$$

$$\lambda_m + \sum_{i=k-m}^{k} \alpha_i^{[m+i]}\, \lambda_{m+i} = \sum_{i=m+1}^{k-1} \left(\frac{\partial \theta_i}{\partial y_m}\right)^T \lambda_i \tag{2.4e}$$
$$+ J_m^T \cdot \sum_{i=k-m}^{k} h_{m+i}\, \beta_i^{[m+i]}\, \lambda_{m+i}\ ,$$
$$m = k-2, \cdots, 0\ .$$

The gradient of the cost function with respect to the initial conditions is

$$\nabla_{y_{\mathrm{ini}}} \Psi = \left(\frac{\partial \Psi}{\partial y_{\mathrm{ini}}} \right)^T = \lambda_0 \ . \tag{2.5}$$

Proof.

The Lagrangian associated with the optimization problem (1.13) is

$$\begin{aligned} \mathcal{L} &= \Psi - \lambda_0^T \cdot (y_0 - y_{\mathrm{ini}}) - \sum_{n=1}^{k-1} \lambda_n^T \cdot \Big(y_n - \theta_n(y_0, \cdots, y_{n-1}) \Big) \\ &\quad - \sum_{n=k}^{N} \lambda_n^T \left(\sum_{i=0}^{k} \alpha_i^{[n]} y_{n-i} - h_n \sum_{i=0}^{k} \beta_i^{[n]} f_{n-i} \right) \ . \end{aligned} \tag{2.6}$$

A variation δy_{ini} in the initial conditions leads to a variation of the Lagrangian

$$\begin{aligned} \delta\mathcal{L} &= \delta\Psi - \lambda_0^T \cdot (\delta y_0 - \delta y_{\mathrm{ini}}) - \sum_{n=1}^{k-1} \lambda_n^T \cdot \left(\delta y_n - \sum_{m=0}^{n-1} \frac{d\theta_n}{dy_m} \, \delta y_m \right) \\ &\quad - \sum_{i=0}^{k} \sum_{n=k}^{N} \lambda_n^T \cdot \Big(\alpha_i^{[n]} I - h_n \, \beta_i^{[n]} \, J_{n-i} \Big) \cdot \delta y_{n-i} \ . \end{aligned} \tag{2.7}$$

Note that under the constraint that $\{y_0, \cdots, y_N\}$ is the solution of (2.1a)–(2.1c) we have that $\mathcal{L} = \Psi$ and therefore $\delta\mathcal{L} = \delta\Psi$.

With the change of variable $n = m + i$ equation (2.7) becomes

$$\begin{aligned} \delta\mathcal{L} &= \delta\Psi - \lambda_0^T \cdot (\delta y_0 - \delta y_{\mathrm{ini}}) - \sum_{n=1}^{k-1} \lambda_n^T \cdot \left(\delta y_n - \sum_{m=0}^{n-1} \frac{d\theta_n}{dy_m} \, \delta y_m \right) \\ &\quad - \sum_{i=0}^{k} \sum_{m=k-i}^{N-i} \lambda_{m+i}^T \cdot \Big(\alpha_i^{[m+i]} I - h_{m+i} \, \beta_i^{[m+i]} \, J_m \Big) \cdot \delta y_m \ . \end{aligned} \tag{2.8}$$

Change of the summation order in the third term rearranges it as

$$\sum_{n=1}^{k-1} \sum_{m=0}^{n-1} = \sum_{m=0}^{k-2} \sum_{n=m+1}^{k-1}$$

Change of the summation order in the last term rearranges it as

$$\sum_{i=0}^{k} \sum_{m=k-i}^{N-i} = \sum_{m=0}^{k-1} \sum_{i=k-m}^{k} + \sum_{m=k}^{N-k} \sum_{i=0}^{k} + \sum_{m=N-k+1}^{N} \sum_{i=0}^{N-m}$$

With this equation (2.8) reads

$$
\begin{array}{rcl}
\delta\mathcal{L} & = & \delta\Psi + \lambda_0^T\,\delta y_{\rm ini} \\
 & & -\delta y_0^T \cdot \left(\lambda_0 - \displaystyle\sum_{i=1}^{k-1} \left(\frac{d\theta_i}{dy_0} \right)^T \lambda_i + \left(\alpha_k^{[k]} I - h_k\, \beta_k^{[k]}\, J_0^T \right) \cdot \lambda_k \right) \\
 & & - \displaystyle\sum_{m=1}^{k-2} \delta y_m^T \cdot \left(\lambda_m - \displaystyle\sum_{i=m+1}^{k-1} \left(\frac{\partial\theta_i}{\partial y_m} \right)^T \lambda_i + \displaystyle\sum_{i=k-m}^{k} \left(\alpha_i^{[m+i]} I - h_{m+i}\, \beta_i^{[m+i]}\, J_m^T \right) \cdot \lambda_{m+i} \right) \\
 & & -\delta y_{k-1}^T \cdot \left(\lambda_{k-1} + \displaystyle\sum_{i=1}^{k} \left(\alpha_i^{[k-1+i]} I - h_{k-1+i}\, \beta_i^{[k-1+i]}\, J_{k-1}^T \right) \cdot \lambda_{k-1+i} \right) \\
 & & - \displaystyle\sum_{m=k}^{N-k} \delta y_m^T \cdot \displaystyle\sum_{i=0}^{k} \left(\alpha_i^{[m+i]} I - h_{m+i}\, \beta_i^{[m+i]}\, J_m^T \right) \cdot \lambda_{m+i} \\
 & & - \displaystyle\sum_{m=N-k+1}^{N} \delta y_m^T \cdot \displaystyle\sum_{i=0}^{N-m} \left(\alpha_i^{[m+i]} I - h_{m+i}\, \beta_i^{[m+i]}\, J_m^T \right) \cdot \lambda_{m+i}\ .
\end{array}
\tag{2.9}
$$

Substituting the adjoint variable recurrence (2.4a)–(2.4e) into (2.9) leads to a formulation of the variation $\delta\mathcal{L}$ that does not depend on the tangent linear variables δy_i at time points other than $i = 0$:

$$
\delta\mathcal{L} = \delta\Psi + \delta y_{\rm ini}^T\,\lambda_0 - \delta y_N^T \left(\frac{\partial g}{\partial y}\,(y_N) \right)^T \quad , \quad \forall\ \delta y_{\rm ini} \in \Re^d\ .
$$

Since under the model constrains $\delta\mathcal{L} = \delta\Psi$ the gradient of Ψ with respect to the initial conditions is

$$
\frac{\partial\Psi}{\partial y_{\rm ini}} \cdot \delta y_{\rm ini} = \frac{\partial\Psi}{\partial y_N} \cdot \delta y_N = \left(\frac{\partial g}{\partial y}\,(y_N) \right) \cdot \delta y_N = \lambda_0^T\,\delta y_{\rm ini} \quad \Rightarrow \quad \left(\frac{\partial\Psi}{\partial y_{\rm ini}} \right)^T = \lambda_0\ .
$$

□

The original LMM method (2.1a)–(2.1c) applied to solve the adjoint ODE reads

$$
\overline{\lambda}_N = \left(\frac{\partial g}{\partial y}\big(y(t_N)\big) \right)^T \ , \tag{2.10a}
$$

$$
\overline{\lambda}_m = \theta_m\left(\overline{\lambda}_N, \cdots, \overline{\lambda}_{m+1} \right)\ , \quad m = N-1, \cdots, N-k+1\ , \tag{2.10b}
$$

$$
\sum_{i=0}^{k} \overline{\alpha}_i^{[m]} \overline{\lambda}_{m+i} = h_{m+1} \sum_{i=0}^{k} \overline{\beta}_i^{[m]} J^T\big(y(t_{m+i})\big) \cdot \overline{\lambda}_{m+i}\ , \quad m = N-k, \cdots, 0 \tag{2.10c}
$$

The coefficients $\overline{\alpha}_i^{[n]}$, $\overline{\beta}_i^{[n]}$ depend on the sequence of steps h_m in reverse order, therefore they depend on the ratios ω_{n+k}^{-1}, $\cdots$, ω_{n+2k-2}^{-1}. They are in general different than the forward method coefficients $\alpha_i^{[n]}$, $\beta_i^{[n]}$ which depend on the ratios ω_n, $\cdots$, ω_{n-k+2}.

The one-leg counterpart [5, Section V.6] of the LMM method (2.10c) is

$$\begin{aligned}
\overline{\lambda}_N &= \left(\frac{\partial g}{\partial y}\big(y(t_N)\big)\right)^T \ , && (2.11a)\\
\overline{\lambda}_m &= \theta_m\left(\overline{\lambda}_N, \cdots, \overline{\lambda}_{m+1}\right) \ , \quad m = N-1, \cdots, N-k+1 \ , && (2.11b)\\
\sum_{i=0}^{k} \overline{\alpha}_i^{[m]} \overline{\lambda}_{m+i} &= h_{m+1}\, J^T\big(y(\tau^{[m]})\big) \cdot \sum_{i=0}^{k} \overline{\beta}_i^{[m]}\, \overline{\lambda}_{m+i} \ , && (2.11c)\\
\tau^{[m]} &= \sum_{\ell=0}^{k} \frac{\overline{\beta}_\ell^{[m]}}{\overline{\beta}^{[m]}}\, t_{m+\ell} \ , \quad \overline{\beta}^{[m]} = \sum_{\ell=0}^{k} \overline{\beta}_\ell^{[m]} \ , &&\\
m &= N-k, \cdots, 0 \ . &&
\end{aligned}$$

Note that, due to linearity of the right hand side, the scaling by the $\overline{\beta}^{[m]}$ does not appear in the sum of $\overline{\lambda}$'s multiplied by J^T. The order of accuracy of the discretization (2.11c) is at most $r+1$, where r is the interpolation order of the method [5, Section V.6]. The interpolation order is the smallest integer r such that for any smooth function $q(t)$ it holds that

$$q\left(\tau^{[m]}\right) - \sum_{\ell=0}^{k} \frac{\overline{\beta}_\ell^{[m]}}{\overline{\beta}^{[m]}}\, q\left(t_{m+i}\right) = \mathcal{O}\left(h^{r+1}\right) \ . \tag{2.12}$$

The discrete adjoint step (2.4c) can be written in the equivalent form

$$\begin{aligned}
\sum_{i=0}^{k} \alpha_i^{[m+i]}\, \lambda_{m+i} &= h_{m+1}\, J^T\big(y_m\big) \cdot \sum_{i=0}^{k} \frac{h_{m+i}}{h_{m+1}}\, \beta_i^{[m+i]}\, \lambda_{m+i}\\
&= h_{m+1}\, J^T\big(y_m\big) \cdot \sum_{i=0}^{k} \widehat{\beta}_i^{[m+i]}\, \lambda_{m+i} \qquad\qquad (2.13)\\
\widehat{\beta}_0^{[m]} &= \omega_{m+1}^{-1}\, \beta_0^{[m]} \ , \quad \widehat{\beta}_1^{[m+1]} = \beta_1^{[m+1]} \ ,\\
\widehat{\beta}_i^{[m+i]} &= \left(\prod_{\ell=2}^{i} \omega_{m+\ell}\right) \beta_i^{[m+i]} \ , \quad i = 2, \cdots, k \ .
\end{aligned}$$

The form (2.13) looks like the one-leg method (2.11c) associated with the LMM. The argument at which the Jacobian is evaluated is, however, different. The initialization of the discrete adjoint (2.4a)–(2.4b) and of the one-leg continuous adjoint (2.11a)–(2.11b) are also different. Moreover the termination relations for the discrete adjoint calculation (2.4d), (2.4e) are different and depend on the initialization procedure of the forward method. We will analyze the impact of these differences on the accuracy of the the discrete adjoint as a numerical method to solve the adjoint ODE.

Example. Consider the variable step BDF2 method [5, Section III.5, page 401] initialized with the backward Euler method:

$$\begin{aligned}
y_1 &= y_0 + h_1\, f_1 \ , \qquad\qquad (2.14)\\
y_n &= \frac{\left(1+\omega_n\right)^2}{1+2\omega_n}\, y_{n-1} - \frac{\omega_n^2}{1+2\omega_n}\, y_{n-2} + h_n\, \frac{1+\omega_n}{1+2\omega_n}\, f_n \ , \quad n = 2, \cdots, N \ .
\end{aligned}$$

For stability we restrict $\omega_n < 1+\sqrt{2}$ [5, Section III.5]. We have that

$$\left(\frac{d\theta_1}{dy_0}\right)^T = \left(I - h_1\, J_1^T\right)^{-1}$$

and the discrete BDF2 adjoint is

$$
\begin{aligned}
\lambda_N &= \left(I - h_N \, \frac{1+\omega_N}{1+2\omega_N} \, J_N^T\right)^{-1} \left(\frac{\partial g}{\partial y}\,(y_N)\right)^T \;, && \text{(2.15a)}\\
\lambda_{N-1} &= \left(I - h_{N-1} \, \frac{1+\omega_{N-1}}{1+2\omega_{N-1}} J_{N-1}^T\right)^{-1} \cdot \left(\frac{(1+\omega_N)^2}{1+2\omega_N}\,\lambda_N\right) \;, && \text{(2.15b)}\\
\lambda_m &= \left(I - h_m \, \frac{1+\omega_m}{1+2\omega_m} \, J_m^T\right)^{-1} \left(\frac{(1+\omega_{m+1})^2}{1+2\omega_{m+1}}\,\lambda_{m+1} - \frac{\omega_{m+2}^2}{1+2\omega_{m+2}}\,\lambda_{m+2}\right) && \text{(2.15c)}\\
&= \left(I - h_{m+1} \, \frac{1+\omega_m}{\omega_{m+1}(1+2\omega_m)} \, J_m^T\right)^{-1} \left(\frac{(1+\omega_{m+1})^2}{1+2\omega_{m+1}}\,\lambda_{m+1} - \frac{\omega_{m+2}^2}{1+2\omega_{m+2}}\,\lambda_{m+2}\right)\\
&\qquad m = N-2, \cdots, 2 \;,\\
\lambda_1 &= \frac{(1+\omega_2)^2}{1+2\omega_2}\,\lambda_2 - \frac{\omega_3^2}{1+2\omega_3}\,\lambda_3 && \text{(2.15d)}\\
\lambda_0 &= \left(I - h_1\, J_1^T\right)^{-1} \lambda_1 - \frac{\omega_2^2}{1+2\omega_2}\,\lambda_2 \;. && \text{(2.15e)}
\end{aligned}
$$

The BDF2 method (2.14) applied to the adjoint ODE leads to the continuous adjoint solution:

$$
\begin{aligned}
&\overline{\lambda}_N = \left(\frac{\partial g}{\partial y}\big(y(t_N)\big)\right)^T \;, \quad \overline{\lambda}_{N-1} = \Big(I - h_N\, J^T\big(y(t_{N-1})\big)\Big)^{-1} \overline{\lambda}_N \;, && \text{(2.16)}\\
&\overline{\lambda}_m = \left(I - h_{m+1}\, \frac{1+\omega_{m+2}^{-1}}{1+2\omega_{m+2}^{-1}}\, J^T\big(y(t_m)\big)\right)^{-1} \left(\frac{\big(1+\omega_{m+2}^{-1}\big)^2}{1+2\omega_{m+2}^{-1}}\,\overline{\lambda}_{m+1} - \frac{\omega_{m+2}^{-2}}{1+2\omega_{m+2}^{-1}}\,\overline{\lambda}_{m+2}\right)\\
&\qquad m = N-2, \cdots, 0 \;.
\end{aligned}
$$

The continuous adjoint coefficients (2.16) depend only on ω_{m+2}^{-1}, while the discrete adjoint coefficients (2.15c) depend on ω_m, ω_{m+1}, ω_{m+2}. The two formulas coincide for constant step sizes ($\omega_m = 1$ for all m).

Example. As a second example consider the variable step AB2 method initialized with the forward Euler method:

$$
\begin{aligned}
y_1 &= y_0 + h_1\, f_0 \;,\\
y_n &= y_{n-1} + h_n\, \frac{2\omega_n+1}{2}\, f_{n-1} - h_n\, \frac{\omega_n}{2}\, f_{n-2} \;, \quad n = 2, \cdots, N \;.
\end{aligned}
\qquad \text{(2.17)}
$$

We have that

$$
\left(\frac{d\theta_1}{dy_0}\right)^T = I + h\, J_0^T
$$

and the discrete AB2 adjoint reads:

$$\lambda_N = \left(\frac{\partial g}{\partial y}\left(y_N\right) \right)^T \;, \tag{2.18a}$$

$$\lambda_{N-1} = \lambda_N + h_N \, \frac{2\omega_N + 1}{2} \, J_{N-1}^T \cdot \lambda_N \;, \tag{2.18b}$$

$$\begin{aligned} \lambda_m &= \lambda_{m+1} + J_m^T \cdot \left(h_{m+1} \, \frac{2\omega_{m+1}+1}{2} \, \lambda_{m+1} - h_{m+2} \, \frac{\omega_{m+2}}{2} \, \lambda_{m+2} \right) \qquad (2.18c) \\ &= \lambda_{m+1} + h_{m+1} \, J_m^T \cdot \left(\frac{2\omega_{m+1}+1}{2} \, \lambda_{m+1} - \frac{\omega_{m+2}^2}{2} \, \lambda_{m+2} \right) \;, \\ &\quad m = N-2, \cdots, 1 \;, \end{aligned}$$

$$\lambda_0 = \lambda_1 + h_1 \, J_0^T \, \lambda_1 - h_2 \, \frac{\omega_2}{2} \, J_0^T \cdot \lambda_2 \;. \tag{2.18d}$$

Solving the adjoint ODE with (2.17) leads to the continuous AB2 adjoint solution

$$\overline{\lambda}_N = \left(\frac{\partial g}{\partial y}\big(y(t_N)\big) \right)^T \;, \quad \overline{\lambda}_{N-1} = \overline{\lambda}_N + h_N \, J^T\big(y(t_N)\big) \, \overline{\lambda}_N \;,$$

$$\overline{\lambda}_m = \overline{\lambda}_{m+1} + h_{m+1} \, \frac{2\omega_{m+2}^{-1}+1}{2} \, J^T\big(y(t_{m+1})\big) \, \overline{\lambda}_{m+1} - h_{m+1} \, \frac{\omega_{m+2}^{-1}}{2} \, J^T\big(y(t_{m+2})\big) \, \overline{\lambda}_{m+2} \;.$$

2.1. Consistency Analysis for Fixed Step Sizes. Consider first the case where the multistep method is applied with a fixed step size. With some abuse of notation relative to (2.3) in this section we consider $h_n = h$ for all n. The LMM coefficients are the same for all steps and the discrete adjoint step (2.4c) at the interior trajectory points reads

$$\sum_{i=0}^{k} \alpha_i \, \lambda_{m+i} = h \, J^T\big(y_m\big) \, \sum_{i=0}^{k} \beta_i \, \lambda_{m+i} \;, \quad m = N-k, \cdots, k \;. \tag{2.19}$$

Proposition 2.2 (Fixed stepsize consistency at interior trajectory points). *In the general case equation (2.19) is a first order consistent method for the adjoint ODE. The order of consistency equals that of the one-leg counterpart for LMMs with*

$$\sum_{\ell=1}^{k} \ell \, \beta_\ell = 0. \tag{2.20}$$

Proof. The consistency analysis can be done by direct differentiation. We take an approach that highlights the relation between (2.19) and the one-leg continuous adjoint step (2.11c). If the forward method is convergent of order p

$$y(t_m) - y_m = \mathcal{O}\left(h^p\right) \;.$$

For the smooth forward solution it holds that

$$\tau^{[m]} = t_m + h \sum_{\ell=0}^{k} \frac{\ell \, \beta_\ell}{\beta} \;, \quad \beta = \sum_{\ell=0}^{k} \beta_\ell \;\neq 0, \quad y(\tau^{[m]}) - y(t_m) = \mathcal{O}\left(h \sum_{\ell=0}^{k} \frac{\ell \, \beta_\ell}{\beta} \right) \;.$$

The step (2.19) can be regarded as a perturbation of the one-leg step (2.11c)

$$\sum_{i=0}^{k} \alpha_i \, \lambda_{m+i} = h \, J^T\big(y(\tau^{[m]})\big) \, \sum_{i=0}^{k} \beta_i \, \lambda_{m+i} + \varepsilon_m$$

$$\varepsilon_m = h \left(J^T\big(y_m\big) - J^T\big(y(\tau^{[m]})\big) \right) \, \sum_{i=0}^{k} \beta_i \, \lambda_{m+i}$$

Since the Jacobian is continuously differentiable we have that

$$\left\| J^T\big(y(\tau^{[m]})\big) - J^T\big(y_m\big) \right\| = \mathcal{O}\left(\|y(\tau^{[m]}) - y_m\|^2 \right)$$

$$y(\tau^{[m]}) - y_m = \underbrace{y(\tau^{[m]}) - y(t_m)}_{\left(\sum_{\ell=0}^{k} \ell \, \beta_\ell\right) \cdot \mathcal{O}(h)} + \underbrace{y(t_m) - y_m}_{\mathcal{O}(h^p)}$$

Under the smoothness assumptions all derivatives are bounded and we have that

$$\varepsilon_m = \left(\sum_{\ell=0}^{k} \ell \, \beta_\ell \right) \cdot \mathcal{O}\left(h^2\right) + \mathcal{O}\left(h^{p+1}\right)$$

The order of consistency of the discrete adjoint step (2.19) is therefore equal to one in the general case, and is equal to the order of consistency of the associated one-leg method when (2.20) holds. For Adams methods the order of consistency of the discrete adjoint is one. For BDF methods $\beta_0 \neq 0$ and $\beta_\ell = 0$, $\ell \geq 1$, therefore the order of consistency equals that of the one-leg counterpart, i.e., equals that of the original method.

□

We are next concerned with the effects of the initialization steps (2.4a), (2.4b) and of the termination steps (2.4d) and (2.4e).

Proposition 2.3 (Accuracy of the adjoint initialization steps).

For a general LMM the discrete adjoint initialization steps (2.4a), (2.4b) do not provide consistent approximations of the adjoint ODE solution. For Adams methods the initialization steps are $\mathcal{O}(h)$ approximations of the continuous adjoint solution.

Proof. Consider without loss of generality that the coefficients are scaled such that $\alpha_0 = 1$. The initialization steps (2.4a), (2.4b) for constant step sizes read:

$$\lambda_N = \left(I - h \, \beta_0 \, J_N^T\right)^{-1} \left(\frac{\partial g}{\partial y}\left(y_N\right) \right)^T \,, \tag{2.21a}$$

$$\sum_{i=0}^{N-m} \alpha_i \, \lambda_{m+i} = h \, J_m^T \cdot \sum_{i=0}^{N-m} \beta_i \, \lambda_{m+i} \,, \quad m = N-1, \cdots, N-k+1 \,. \tag{2.21b}$$

Note that if the forward method converges at order p

$$\overline{\lambda}(t_N) = \left(\frac{\partial g}{\partial y}\big(y(t_N)\big) \right)^T = \left(\frac{\partial g}{\partial y}\big(y_N\big) \right)^T + \mathcal{O}\left(h^p\right) \,.$$

For implicit schemes ($\beta_0 \neq 0$) equation (2.21a) gives $\lambda_N = \overline{\lambda}(t_N - \beta_0 h) + \mathcal{O}\left(h^2\right)$ $= \overline{\lambda}(t_N) + \mathcal{O}\left(h\right)$.

Under the smoothness assumptions the forward solution $y(t)$ can be extended uniquely beyond t_{end} for all times $t < t_{\text{end}} + \Delta t$. Under the smoothness assumptions the continuous adjoint solution $\overline{\lambda}(t)$ can also be extended uniquely for all times $t < t_{\rm end} + \Delta t$ (with eventually a redefinition of Δt). Choose a small enough time step such that $k\,h < \Delta t$ and define $\lambda_i = \overline{\lambda}(t_i)$ for $N \le i \le N+k-1$. Equation (2.21b) can be written as the discrete adjoint formula at internal time steps plus a perturbation:

$$\sum_{i=0}^{k} \alpha_i\,\lambda_{m+i} = h\,J_m^T \cdot \sum_{i=0}^{k} \beta_i\,\lambda_{m+i} + \varepsilon_m\ , \quad m = N-1, \cdots, N-k+1\ ,$$

$$\varepsilon_m = \sum_{i=N-m+1}^{k} \alpha_i\,\overline{\lambda}(t_{m+i}) - h\,J_m^T \cdot \sum_{i=N-m+1}^{k} \beta_i\,\overline{\lambda}(t_{m+i})\ .$$

The perturbation term is $\|\varepsilon_m\| = \mathcal{O}(1)$ in the general case where at least one of α_2, $\cdots$, α_k is nonzero. For Adams methods where $\alpha_2 = \cdots = \alpha_k = 0$ the magnitude of the perturbation term is $\|\varepsilon_m\| = \mathcal{O}(h)$. □

Proposition 2.4 (Accuracy of the adjoint termination steps).

For a general LMM the discrete adjoint termination steps (2.4d) and (2.4e) are not consistent discretizations of the adjoint ODE.

Proof. With the coefficients scaled such that $\alpha_0 = 1$ the termination step (2.4d) can be written as

$$\sum_{i=0}^{k} \alpha_i\,\lambda_{k-1+i} = h\,J_{k-1}^T \cdot \sum_{i=0}^{k} (\beta_i\,\lambda_{k-1+i}) + \varepsilon_{k-1}$$

$$\varepsilon_{k-1} = -h\,\beta_0\,J_{k-1}^T \cdot \lambda_{k-1}$$

This is the discrete adjoint formula at internal step $(k-1)$ plus an $\mathcal{O}(h)$ perturbation added in the case of implicit methods.

The termination steps (2.4e) for constant step sizes depend on the particular initialization procedure of the forward method

$$\lambda_m + \sum_{i=k-m}^{k} \alpha_i\,\lambda_{m+i} = \sum_{i=m+1}^{k-1} \left(\frac{\partial \theta_i}{\partial y_m}\right)^T \lambda_i + h\,J_m^T \cdot \sum_{i=k-m}^{k} \beta_i\,\lambda_{m+i}\ ,$$

$$m = k-2, \cdots, 1\ ,$$

$$\lambda_0 + \alpha_k\,\lambda_k = \sum_{i=1}^{k-1} \left(\frac{d\theta_i}{dy_0}\right)^T \lambda_i + h\,J_0^T \cdot (\beta_k\,\lambda_k)\ .$$

In general these are not consistent discretizations of the adjoint ODE at the corresponding times, as will be seen in two examples presented next.
□

Example. Consider the fixed-step Adams-Bashforth method AB2 (2.17) started with an explicit Euler step and the corresponding discrete adjoint process (2.18a)–(2.18d). For the fixed step formulas one replaces $h_n = h$ and $\omega_n = 1$ for all n in (2.17) and (2.18a)–(2.18d).

The initialization step (2.18a) gives $\lambda_N = \overline{\lambda}(t_N) + \mathcal{O}(h^p)$ (with the error coming from the fact that the argument of $\partial g/\partial y$ is the numerical and not the exact solution at t_N). The step (2.18b) provides an approximation $\lambda_{N-1} = \overline{\lambda}(t_{N-1} - h/2) + \mathcal{O}(h^2)$ $= \overline{\lambda}(t_{N-1}) + \mathcal{O}(h)$.

If $\lambda_2 = \overline{\lambda}(t_2)$ and $\lambda_1 = \overline{\lambda}(t_1)$ the termination step (2.18d) provides an approximation $\lambda_0 = \overline{\lambda}(t_0) + \mathcal{O}(h)$.

Example. Consider next the fixed step BDF2 method (2.14) started with an implicit Euler step and the corresponding discrete adjoint process (2.15a)–(2.15e). For the fixed step formulas one replaces $h_n = h$ and $\omega_n = 1$ for all n in (2.14) and (2.15a)–(2.15e). The initialization step (2.15a) computes $\lambda_N = \overline{\lambda}(t_N - 2h/3) + \mathcal{O}(h^2) = \overline{\lambda}(t_N) + \mathcal{O}(h)$. The step (2.15b) provides $\lambda_{N-1} = (4/3)\overline{\lambda}(t_{N-1} - h/3) + \mathcal{O}(h^2) = \overline{\lambda}(t_{N-1}) + \mathcal{O}(1)$.

If $\lambda_2 = \overline{\lambda}(t_2)$ and $\lambda_3 = \overline{\lambda}(t_3)$ the termination relation (2.15d) provides $\lambda_1 = \overline{\lambda}(t_2 + h/2) + \mathcal{O}(h^2) = \overline{\lambda}(t_1) + \mathcal{O}(h)$. If in addition $\lambda_1 = \overline{\lambda}(t_1)$ then (2.15e) computes $\lambda_0 = \overline{\lambda}(t_0) + \mathcal{O}(1)$.

In summary the discrete adjoints of the fixed-step LMM are at least first order consistent with the adjoint ODE at the intermediate points. The initialization and the termination relations introduce perturbations that can be as large as $\mathcal{O}(1)$. One can change the discrete adjoint initialization and the termination steps to consistent relations. In this case we expect the method to be at least first order consistent with the adjoint ODE.

2.2. Consistency Analysis for Variable Step Sizes. For variable steps the consistency of the discrete adjoint with the adjoint ODE is not automatic. In this section we will use the notation (2.2).

Proposition 2.5 (Variable stepsize consistency at the intermediate trajectory points).

In general the numerical process (2.4a)–(2.4e) is not a consistent discretization of the adjoint ODE (1.9).

Proof. The consistency can be lost due to the fact that the coefficients $\alpha_i^{[m+i]}$, $\beta_i^{[m+i]}$ in (2.4c) correspond to different time steps in the forward integration (2.1c).

The relation (2.4c) can be regarded as a one-leg discretization method (2.13) applied to the adjoint ODE. Replacing $J^T\big(y_m\big)$ by $J^T\big(y(t_m)\big)$ in (2.13) introduces an $\mathcal{O}(h^{p+1})$ approximation error

$$\sum_{i=0}^{k} \alpha_i^{[m+i]}\, \lambda_{m+i} = h_{m+1}\, J^T\big(y(t_m)\big) \cdot \sum_{i=0}^{k} \widehat{\beta}_i^{[m+i]}\, \lambda_{m+i} + \mathcal{O}\big(h^{p+1}\big)\ , \quad m = N-k, \cdots, k\ .$$

The following consistency analysis of (2.13) will be performed on this modified equation and its results are valid within $\mathcal{O}(h^{p+1})$.

To obtain consistency conditions we substitute a smooth local solution $q(t_\ell)$ for λ_ℓ, $-q'(t_\ell)$ for $J^T\big(y(t_\ell)\big)\lambda_\ell$, and expand in Taylor series around t_m.

The zeroth order consistency condition reads

$$\sum_{i=0}^{k} \alpha_i^{[m+i]} = 0\ . \tag{2.23}$$

For a general sequence of step sizes h_m the values of $\alpha_i^{[m+i]}$ at different steps m are not necessarily constrained by (2.23). A general discrete adjoint LMM process is therefore inconsistent with the adjoint ODE.

In the case where the forward steps are chosen automatically to maintain the local error estimate under a given threshold the step changes are smooth [4, Section III.5] in the sense that

$$|\omega_n - 1| \le \text{const} \cdot h_{n-1} \quad \Rightarrow \quad \omega_n = 1 + \mathcal{O}(|h|)\ . \tag{2.24}$$

Recall that we do not consider the derivatives of the step sizes with respect to system state. Nevertheless, let us look at the impact of these smooth step changes on the discrete adjoint consistency. If the LMM coefficients $\alpha_i^{[n]}$ and $\beta_i^{[n]}$ depend smoothly on step size ratios ω_n, then for each n they are small perturbations of the constant step size values: $\alpha_i^{[n]} = \alpha_i + \mathcal{O}(|h|)$ and $\beta_i^{[n]} = \beta_i + \mathcal{O}(|h|)$. It then holds that $\sum_{i=0}^{k} \alpha_i^{[m+i]} = \mathcal{O}(|h|)$. Consequently the zeroth order consistency condition (2.23) is satisfied. The $\mathcal{O}(|h|)$ perturbation, however, prevents first order consistency of the discrete adjoint method.

For Adams methods in particular the relation (2.23) is automatically satisfied. The first order consistency condition for Adams methods reads

$$\sum_{i=0}^{k} \widehat{\beta}_i^{[m+i]} = 1 \ . \tag{2.25}$$

For a general sequence of step sizes h_m the values of $\beta_i^{[m+i]}$ at different steps m are not constrained by any relation among them and (2.25) is not satisfied.

If the forward steps are chosen such that (2.24) holds [4, Section III.5], and if the LMM coefficients depend smoothly on step size ratios, we have that $\sum_{i=0}^{k} \widehat{\beta}_i^{[m+i]} = 1 + \mathcal{O}(|h|)$. In this situation the discrete Adams adjoint methods are first order consistent with the adjoint ODE.

□

Example. Consider the variable step BDF2 method (2.14) and its discrete adjoint (2.15c). The zeroth consistency condition (2.23)

$$\sum_{i=0}^{k} \alpha_i^{[m+i]} = 1 - \frac{\left(1+\omega_{m+1}\right)^2}{1+2\omega_{m+1}} + \frac{\omega_{m+2}^2}{1+2\omega_{m+2}} = 0$$

has a single positive solution $\omega_{m+2} = \omega_{m+1} = \omega$ (the step sizes change at a constant ratio). Substituting this into the first order consistency condition (2.25) leads to the unique solution $\omega = 1$ (constant step sizes).

Example. As a second example consider the variable step AB2 method (2.17) and its discrete adjoint. The method is zeroth order consistent. The first order condition (2.25) is

$$2\omega_{m+1} = \omega_{m+2}^2 + 1$$

and holds only when the forward time steps follow a certain pattern (all the step ratios are determined by the last ω_N: $\omega_{N-1} = (\omega_N^2 + 1)/2$, $\omega_{N-2} = (\omega_{N-1}^2 + 1)/2$, etc.)

3. Stability considerations. The method (2.1a)–(2.1c) is zero-stable if it has only bounded solutions when applied to the test problem

$$y' = 0 \ , \quad y(t_{\mathrm{ini}}) = y_{\mathrm{ini}} \ , \quad t_{\mathrm{ini}} \le t \le t_{\mathrm{end}} \ . \tag{3.1}$$

To be specific consider the LMM (2.1a)–(2.1c) scaled such that $\alpha_0^{[n]} = 1$ for all n. Using the notation ${}_1 = [1, 0, \cdots, 0]^T$, $\;= [1, 1, \cdots, 1]^T$, and

$$Y_n = \begin{bmatrix} y_n \\ \vdots \\ y_{n-k+1} \end{bmatrix} \ , \quad A_n = \begin{bmatrix} -\alpha_1^{[n]} I & \cdots & -\alpha_{k-1}^{[n]} I & -\alpha_k^{[n]} I \\ I & & 0 & 0 \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & I & 0 \end{bmatrix} \ ,$$

the LMM applied to the test problem (3.1) can be written in the equivalent one-step form

$$\begin{aligned} Y_{k-1} &= (\quad \otimes I) \cdot y_{\rm ini} \ , & (3.2a)\\ Y_n &= A_n\, Y_{n-1} \ , \quad n = k, \cdots, N \ , & (3.2b)\\ y_N &= \left({}^T_1 \otimes I \right) \cdot Y_N \ , \quad \Psi = g\big(y_N\big) \ . & (3.2c) \end{aligned}$$

The LMM (3.2a)–(3.2b) (i.e., (2.1a)–(2.1c)) is zero-stable if [4, Definition 5.4]

$$\| A_{n+\ell}\, A_{n+\ell-1} \, \cdots \, A_{n+1}\, A_n \| \le \text{const} \quad \forall \; n, \ell > 0 \ . \tag{3.3}$$

A consequence of zero-stability (3.3) is that small changes $\delta y_{\rm ini}$ in the initial conditions of the test problem lead to small changes $\delta \Psi$ in the cost function since

$$\delta \Psi = \frac{\partial g}{\partial y}\,(y_N) \cdot \left({}^T_1 \otimes I \right) \cdot \left(\prod_{n=N}^{k} A_n \right) \cdot (\quad \otimes I) \cdot \delta y_{\rm ini} \ .$$

The discrete adjoint of the numerical process (3.2a)–(3.2c) is

$$\begin{aligned} \Lambda_N &= (\ {}_1 \otimes I) \, \left(\frac{\partial g}{\partial y}(y_N) \right)^T & (3.4a)\\ \Lambda_{n-1} &= A_n^T\, \Lambda_n \ , \quad n = N, \cdots, k \ , & (3.4b)\\ \lambda_0 &= \left(\ {}^T \otimes I \right) \cdot \Lambda_{k-1} \ . & (3.4c) \end{aligned}$$

This is the equivalent one-step form of the discrete adjoint (2.4a)–(2.4e) on the test problem (3.1). By analogy with the stability of the forward integration, the discrete adjoint process (3.4a)–(3.4c) is zero-stable if

$$\left\| A_n^T\, A_{n+1}^T \, \cdots \, A_{n+\ell-1}^T\, A_{n+\ell}^T \right\| \le \text{const} \quad \forall \; n, \ell > 0 \ , \tag{3.5}$$

which ensures that all its numerical solutions remain bounded. The product of matrices in (3.5) is the transpose of the product of matrices in (3.3), and consequently if (3.3) holds then (3.5) holds. In other words if a variable-step LMM is zero-stable then its discrete adjoint is zero-stable. A consequence of the discrete adjoint zero-stability (3.5) is that small perturbations of the adjoint boundary condition $(\partial g/\partial y)^T\,(y_N)$ lead to only small changes in the adjoint initial value, since

$$\lambda_0 = \left(\ {}^T \otimes I \right) \cdot \left(\prod_{n=k}^{N} A_n^T \right) \cdot (\ {}_1 \otimes I) \cdot \left(\frac{\partial g}{\partial y}(y_N) \right)^T \ .$$

4. Derivatives at the Initial Time. We now prove a remarkable property of the discrete LMM adjoints. Even if the discrete adjoint variables λ_n are poor approximations of the continuous adjoints $\overline{\lambda}(t_n)$ at the intermediate grid points, the discrete adjoint at the initial time converges to the continuous adjoint variable with the same order as the original LMM.

Consider a LMM (2.1a)–(2.1c) that computes numerical approximations of (1.6). The last approximation time $t_N = t_{\rm end}$ is fixed (for any N) and coincides with the predefined ODE final time. We denote the sequence of discretization step sizes and the maximum step size as specified in (2.2) and (2.3). The number of steps depends on the step discretization sequence, $N = N(h)$. The numerical approximations depend on the time discretization grid and are denoted by y_n^h, $n = 0 \cdots N(h)$.

PROPOSITION 4.1 (Consistency at the initial time).

Consider a LMM (2.1a)–(2.1c) convergent of order p, and initialized with linear numerical methods. (This covers the typical situation where the initialization procedures $\theta_1, \cdots, \theta_{k-1}$ are Runge Kutta or linear multistep numerical methods). The numerical solutions at the final time are such that

$$\left\| y_{N(h)}^h - y\left(t_{\mathrm{end}}\right) \right\|_\infty = \mathcal{O}\left(|h|^p\right) \ , \quad \forall h \ : \quad |h| \le H \ ,$$

for a small enough threshold H. Let λ_n^h be the solution of the discrete LMM adjoint process (2.4a)–(2.4e).

Then the discrete adjoint solution λ_0^h is an order p approximation of the continuous adjoint $\lambda(t_0)$ at the initial time, i.e.

$$\left\| \lambda_0^h - \overline{\lambda}\left(t_0\right) \right\|_\infty = \mathcal{O}\left(|h|^p\right) \ , \quad \forall h \ : \quad |h| \le H \ , \tag{4.1}$$

for a small enough threshold H.

Proof. The proof is based on the linearity of the LMM and of its starting procedures, which leads to an order p approximation of the full sensitivity matrix.

Step 1: The continuous sensitivity matrix. The continuous sensitivity matrix $S(t) \in \Re^{d \times d}$ is defined by

$$S^{i,j}(t) = \frac{\partial y^i(t)}{\partial y^j\left(t_{\mathrm{ini}}\right)} \ , \quad 1 \le i, j \le d \ , \quad t_{\mathrm{ini}} \le t \le t_{\mathrm{end}} \ ,$$

and contains the derivatives of the ODE solution components at time t with respect to the initial value components. Superscripts are indices of matrix or vector components.

Infinitesimal perturbations $\delta y(t_{\mathrm{ini}})$ of the initial conditions of the ODE (1.6) lead to infinitesimal changes in the solution

$$\delta y(t) = \frac{\partial y(t)}{\partial y\left(t_{\mathrm{ini}}\right)} \cdot \delta y(t_{\mathrm{ini}}) = S(t) \cdot \delta y(t_{\mathrm{ini}}) \ , \quad t_{\mathrm{ini}} \le t \le t_{\mathrm{end}} \ .$$

If the perturbation affects only the j-th component of the initial vector the solution of (4.2) is the j-th column of the sensitivity matrix $s_j = S^{1:d,j}$

$$\delta y\left(t_{\mathrm{ini}}\right) = \ \ _j \quad \Rightarrow \quad \delta y(t) = S(t) \cdot \ \ _j = s_j(t) = \frac{\partial y(t)}{\partial y^j\left(t_{\mathrm{ini}}\right)} \ , \quad t_{\mathrm{ini}} \le t \le t_{\mathrm{end}} \ .$$

Infinitesimal solution changes $\delta y(t)$ propagate forward in time according to the *tangent linear ODE model*

$$\delta y' = J(y) \cdot \delta y \ , \quad \delta y\left(t_{\mathrm{ini}}\right) = \delta y\left(t_{\mathrm{ini}}\right) \ , \quad t_{\mathrm{ini}} \le t \le t_{\mathrm{end}} \ . \tag{4.2}$$

which is obtained by taking the variation of (1.6).

The entire sensitivity $d \times d$ matrix $S\left(t_{\mathrm{end}}\right)$ can be obtained column by column via d forward solutions of the tangent linear model (4.2) initialized with $s_j\left(t_{\mathrm{ini}}\right) = \ \ _j$. Since the tangent linear model (4.2) depends on the ODE solution (which is an argument of the Jacobian) one has to solve simultaneously for (1.6) and (4.2)

$$\begin{bmatrix} y \\ \delta y \end{bmatrix}' = \begin{bmatrix} f(y) \\ J(y) \cdot \delta y \end{bmatrix} \ , \qquad \begin{bmatrix} y\left(t_{\mathrm{ini}}\right) \\ \delta y\left(t_{\mathrm{ini}}\right) \end{bmatrix} = \begin{bmatrix} y_{\mathrm{ini}} \\ \ \ _j \end{bmatrix} \ , \qquad t_{\mathrm{ini}} \le t \le t_{\mathrm{end}} \ . \tag{4.3}$$

A numerical solution of (4.3) is obtained by applying the LMM method (2.1a)–(2.1c) with discretization steps h to this extended system. The LMM solves simultaneously for the solution $y(t)$ and the sensitivity $\delta y(t)$:

$$y_0 = y_{\rm ini}\ , \quad \delta y_0 = \ {}_j\ , \tag{4.4a}$$

$$\left[\begin{array}{c} y_n \\ \delta y_n \end{array}\right] = \theta_n \left(\left[\begin{array}{c} y_0 \\ \delta y_0 \end{array}\right], \cdots, \left[\begin{array}{c} y_{n-1} \\ \delta y_{n-1} \end{array}\right] \right)\ , \quad n = 1, \cdots, k-1\ , \tag{4.4b}$$

$$\sum_{i=0}^{k} \alpha_i^{[n]}\, y_{n-i} = h_n \sum_{i=0}^{k} \beta_i^{[n]}\, f_{n-i}\ , \tag{4.4c}$$

$$\sum_{i=0}^{k} \alpha_i^{[n]}\, \delta y_{n-i} = h_n \sum_{i=0}^{k} \beta_i^{[n]} J_{n-i} \cdot \delta y_{n-i}\ , \quad n = k, \cdots, N\ . \tag{4.4d}$$

The numerical solutions (4.4a)–(4.4d) at final time are $y^h_{N(h)}$ and $\delta y^h_{N(h)} = (s_j)^h_{N(h)}$, $j = 1, \cdots, d$, where we have explicitly represented the dependency of the solution on the step size sequence. By repeating the numerical integration (4.4a)–(4.4d) with the initial perturbations $\delta y_0 = \ {}_j$, $j = 1, \cdots, d$, and with the same step size sequence h, we obtain numerical approximations $(s_j)^h_{N(h)}$ for each column of the sensitivity matrix, $j = 1, \cdots, d$. The numerically-solved full sensitivity matrix is therefore

$$S^h_{N(h)} = \left[(s_1)^h_{N(h)} \cdots (s_d)^h_{N(h)} \right]\ .$$

The eigenvalues of the Jacobian of the ODE (1.6) and of the Jacobian of the tangent linear ODE are the same. Therefore, for the same sequence of steps h, the stability properties of the LMM applied to solve (1.6) are the same as the stability properties of the numerical process (4.4a)–(4.4d). Since the numerical LMM is convergent with order p, each component of the numerically solved sensitivity matrix converges to the corresponding component of the exact sensitivity matrix with order p

$$\max_{1 \le i,j \le n} \left| \left(S^h_{N(h)} \right)^{i,j} - S^{i,j}\left(t_{\rm end}\right) \right| = \mathcal{O}\left(|h|^p\right)\ , \quad \forall\ h\ :\ \ |h| \le H\ . \tag{4.5}$$

Step 2: The discrete sensitivity matrix. The discrete sensitivity matrix $Q_n \in \Re^{d\times d}$ is defined by

$$\left(Q_n\right)^{i,j} = \frac{\partial y_n^i}{\partial y_0^j}\ , \quad 1 \le i,j \le d\ , \quad 0 \le n \le N\ ,$$

and contains the derivatives of the numerical solution components at (the discrete approximation time) t_n with respect to the initial value components.

An infinitesimal perturbation δy_0 of the initial conditions leads to an infinitesimal change δy_n the numerical solution (2.1a)–(2.1c). Similar to the continuous case, a perturbation $\delta y_0 = \ {}_j$ leads to a solution that is the j-th column of the sensitivity matrix $q_j = Q^{1:d,j}$

$$\delta y_n = Q_n \cdot \delta y_0\ , \quad \delta y_0 = \ {}_j \quad \Rightarrow \quad \delta y_n = \left(q_j\right)_n = \frac{\partial y_n}{\partial y_0^j}\ , \quad 0 \le n \le N\ .$$

The solution changes δy_n propagate in time according to the *tangent linear LMM*, which is obtained by taking the variation of (2.1a)–(2.1c). With $\delta y_0 = {}_j$ and $\delta y_n = (q_j)_n$ we have

$$y_0 = y_{\text{ini}} \ , \quad \delta y_0 = {}_j \ , \tag{4.6a}$$

$$\left[\begin{array}{c} y_n \\ \delta y_n \end{array} \right] = \left[\begin{array}{c} \theta_n\left(y_0, \cdots, y_{n-1}\right) \\ \displaystyle\sum_{p=0}^{n-1} \frac{\partial \theta_n}{\partial y_p}\left(y_0, \cdots, y_{n-1}\right) \delta y_p \end{array} \right] \ , \tag{4.6b}$$

$$n = 1, \cdots, k-1 \ ,$$

$$\sum_{i=0}^{k} \alpha_i^{[n]}\, y_{n-i} = h_n \sum_{i=0}^{k} \beta_i^{[n]}\, f_{n-i} \ , \quad n = k, \cdots, N \ , \tag{4.6c}$$

$$\sum_{i=0}^{k} \alpha_i^{[n]} \delta y_{n-i} = h_n \sum_{i=0}^{k} \beta_i^{[n]}\, J_{n-i} \cdot \delta y_{n-i} \ . \tag{4.6d}$$

The numerical solutions (4.6a)–(4.6d) at final time are $y^h_{N(h)}$ and $\delta y^h_{N(h)} = (q_j)^h_{N(h)}$, $j = 1, \cdots, d$. By repeating (4.6a)–(4.6d) with the initial perturbations $\delta y_0 = {}_j$, $j = 1, \cdots, d$, and with the same step size sequence h, we obtain numerical approximations $(q_j)^h_{N(h)}$. The full discrete sensitivity matrix is therefore

$$Q^h_{N(h)} = \left[(q_1)^h_{N(h)} \cdots (q_d)^h_{N(h)} \right] \ .$$

Let's compare the two numerical processes (4.4a)–(4.4d) and (4.6a)–(4.6d), assuming that both use the same step size sequence h. The initial values (4.4a) and (4.6a) are the same for both processes.

We now use the assumption that the initialization procedures θ_n are linear methods (like Runge Kutta or linear multistep). In this case the initialization procedure applied to the extended system (4.3) is the same as the *variation* of the original initialization procedure applied to (1.6). With some abuse of notation we have that the linearity of the initialization scheme implies that:

$$\theta_n \left(\left[\begin{array}{c} y_0 \\ \delta y_0 \end{array} \right], \cdots, \left[\begin{array}{c} y_{n-1} \\ \delta y_{n-1} \end{array} \right] \right) = \left[\begin{array}{c} \theta_n\left(y_0, \cdots, y_{n-1}\right) \\ \displaystyle\sum_{p=0}^{n-1} \frac{\partial \theta_n}{\partial y_p}\left(y_0, \cdots, y_{n-1}\right) \delta y_p \end{array} \right] \ .$$

Therefore initialization steps (4.6b) are (4.4b) are identical.

The numerical solution y_0, $\cdots$, y_N at all discretization points computed with (4.4b), (4.4b), and (4.4c) is the same as the solution computed with (4.6a), (4.6b), and (4.6c) since the formulas are identical and the sequence of step sizes h is the same.

Furthermore, the LMM discretization of the tangent linear ODE (4.4d) gives the same discrete process as the tangent linear LMM (4.6d). The linearity of the LMM makes it "invariant under differentiation". Consequently, the numerical sensitivity matrix (4.6a)–(4.6d) is the same as the numerical approximation of the ODE sensitivity matrix (4.4a)–(4.4d),

$$Q^h_{N(h)} = S^h_{N(h)} \ . \tag{4.7}$$

From (4.5) and (4.7) we conclude that, in the infinity matrix norm,

$$\left\| Q^h_{N(h)} - S\left(t_{\text{end}}\right) \right\|_\infty = \mathcal{O}\big(|h|^p\big) \ , \quad \forall\ h \ : \ |h| \le H \ , \tag{4.8}$$

and

$$\left\| Q^h_{N(h)} \right\|_\infty \le \| S\,(t_{\rm end}) \|_\infty + \mathcal{O}\big(|h|^p\big) = \mathcal{O}\big(1\big) \ , \quad \forall\ h\ :\ \ |h| \le H\ . \tag{4.9}$$

Step 3: The adjoint approximation error at the initial time. The continuous adjoint variable at the initial time is

$$\overline{\lambda}\,(t_{\rm ini}) = \left(\frac{\partial g\big(y(t_{\rm end})\big)}{\partial y\,(t_{\rm ini})} \right)^T = \left(\frac{\partial g}{\partial y}\big(y(t_{\rm end})\big) \cdot \frac{\partial y(t_{\rm end})}{\partial y\,(t_{\rm ini})} \right)^T = S^T(t_{\rm end}) \cdot \left(\frac{\partial g}{\partial y}\big(y(t_{\rm end})\big) \right)^T \ .$$

The discrete adjoint variable at the initial time is

$$\lambda_0 = \left(\frac{\partial g\left(y^h_{N(h)}\right)}{\partial y_0} \right)^T = \left(\frac{\partial g}{\partial y}\left(y^h_{N(h)}\right) \cdot \frac{\partial y^h_{N(h)}}{\partial y_0} \right)^T = \left(Q^h_{N(h)}\right)^T \cdot \left(\frac{\partial g}{\partial y}\left(y^h_{N(h)}\right) \right)^T \ .$$

The adjoint difference is

$$\begin{aligned} \overline{\lambda}\,(t_{\rm ini}) - \lambda_0 &= S^T(t_{\rm end}) \cdot \left(\frac{\partial g}{\partial y}\left(y(t_{\rm end})\right) \right)^T - \left(Q^h_{N(h)}\right)^T \cdot \left(\frac{\partial g}{\partial y}\left(y^h_{N(h)}\right) \right)^T \\ &= \left(S(t_{\rm end}) - Q^h_{N(h)} \right)^T \cdot \left(\frac{\partial g}{\partial y}\left(y(t_{\rm end})\right) \right)^T \\ &\quad + \left(Q^h_{N(h)}\right)^T \cdot \left(\frac{\partial g}{\partial y}\left(y(t_{\rm end})\right) - \frac{\partial g}{\partial y}\left(y^h_{N(h)}\right) \right)^T \ . \end{aligned} \tag{4.10}$$

Since g is smooth and the LMM converges at order p

$$\left\| \frac{\partial g}{\partial y}\left(y(t_{\rm end})\right) - \frac{\partial g}{\partial y}\left(y^h_{N(h)}\right) \right\|_\infty = \mathcal{O}\left(\left\| y(t_{\rm end}) - y^h_{N(h)} \right\|_\infty \right) = \mathcal{O}\left(|h|^p\right) \ , \quad \forall\ h\ :\ \ |h| \le H\ .$$

Taking infinity norms in (4.10), using (4.9) and the fact that $\|\partial g/\partial y(y(t_{\rm end}))\|_\infty$ is independent of the discretization h leads to the bound (4.1).
□

Comment. The crucial property used in the proof of Proposition 4.1 is the linearity of the method and its initialization procedure, which makes the tangent linear LMM to be the same as the LMM applied to solve the tangent linear ODE. The tangent linear LMM solves the entire sensitivity matrix as accurately as it solves for the solution.

5. Numerical Experiments. We illustrate the theoretical findings with numerical results on the Arenstorf orbit [4], which is define by the following ODE:

$$\begin{aligned} \left(y^1\right)' &= y^3\ , \quad \left(y^2\right)' = y^4 \\ \left(y^3\right)' &= y^1 + 2y^4 - \hat{\mu}\,\frac{y^1+\mu}{\left[(y^1+\mu)^2+(y^2)^2\right]^{3/2}} - \mu\frac{y^1-\hat{\mu}}{\left[(y^1-\hat{\mu})^2+(y^2)^2\right]^{3/2}} \\ \left(y^4\right)' &= y^2 - 2y^3 - \hat{\mu}\,\frac{y^2}{\left[(y^1+\mu)^2+(y^2)^2\right]^{3/2}} - \mu\frac{y^2}{\left[(y^1-\hat{\mu})^2+(y^2)^2\right]^{3/2}}\ , \end{aligned} \tag{5.1}$$

where $\mu = 0.012277471$ and $\hat{\mu} = 1 - \mu$. The integration time interval is $t_0 = 0$ to $t_{\rm end} = 1$ and the initial conditions are:

$$y^1(t_0) = 0.994\ , \quad y^2(t_0) = y^3(t_0) = 0\ , \quad y^4(t_0) = -2.0016\ .$$

We consider the adjoints of the cost functional

$$\Psi = g\big(y(t_{\rm end})\big) = y^1\,(t_{\rm end}) \quad \text{where} \quad \left(\frac{\partial g}{\partial y}\big(y(t_{\rm end})\big)\right)^T = e_1 \ .$$

For the integration we choose the explicit Adams-Bashforth methods of order two (AB2) and three (AB3) and the second order BDF2 method. AB2 is initialized with the forward Euler method, AB3 is initialized with a second order explicit Runge Kutta method, and BDF2 is initialized with the implicit Euler method. This allows each method to converge at the theoretical order. The simulations are performed in Matlab. The reference solutions for the Arenstorf system and its continuous adjoint ODE are obtained with the ode45 routine with the tight tolerances RelTol = 1.e-8, AbsTol = 1.e-8. The RMS norms of the difference between the numerical adjoint solution $(\lambda_n)_{\rm num}$ and the reference continuous adjoint solution $(\overline{\lambda}_n)_{\rm ref}$ at each time moment define instantaneous errors

$$E_n = \sqrt{\frac{1}{d}\sum_{i=1}^{d}\left(\frac{(\lambda_n^i)_{\rm num} - (\overline{\lambda}_n^i)_{\rm ref}}{(\lambda_n^i)_{\rm ref}}\right)^2} \ , \quad n = 0, \cdots, N \ . \qquad (5.2)$$

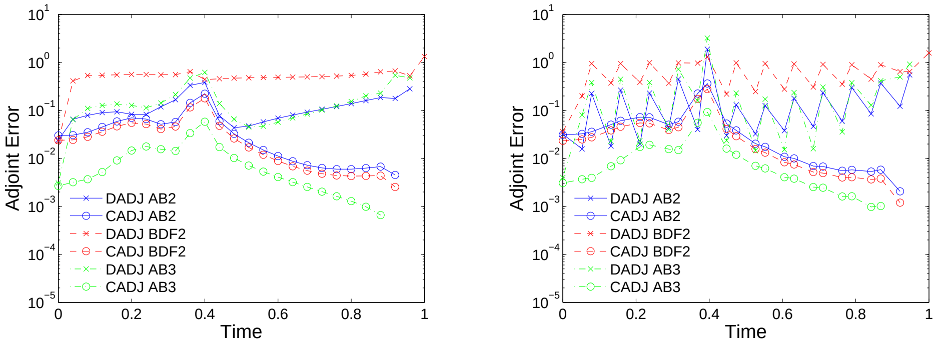
As seen in Section 4 of special interest are the initial time adjoint errors E_0. The trajectory errors measure the total difference between the numerical and the reference adjoint solutions throughout the integration interval:

$$\|E\| = \sqrt{\frac{1}{N+1}\sum_{n=0}^{N} E_n^2} \ . \qquad (5.3)$$

Figure 5.1 shows the time evolution of instantaneous errors (5.2) for the continuous and discrete adjoint solutions of AB2, AB3, and BDF2 with $N = 25$ integration time steps. The fixed step solutions use a step size $h = (t_{\rm end} - t_{\rm ini})/N$. The variable step sizes alternate between small even numbered steps and large odd numbered steps: $\Delta t = (t_{\rm end} - t_{\rm ini})/(1.5\,N)$ and $h_1 = 2\Delta t$, $h_2 = \Delta t$, $h_3 = 2\Delta t$, $h_4 = \Delta t$, etc. The average step size is the same in both situations. The discrete BDF2 solution has the largest errors at the internal points, but note the relatively small error at the initial time. A comparison of Figure 5.1(a) with Figure 5.1(b) reveals that the continuous adjoint errors are relatively unaffected by the variation of step sizes. However, the errors in the discrete AB2, AB3, and BDF2 adjoints show a "zig-zag" pattern that follows the pattern of step size changes.

We next investigate the decrease in error magnitude with increasing number of steps for variable step size integration. Figure 5.2 shows the work-precision diagrams for variable step size integration. The AB2 and BDF2 continuous adjoint trajectory errors (5.3) in Figure 5.2(a) decrease at a second order rate, while the AB3 continuous adjoint error decreases at a third order rate. The AB2 and AB3 discrete adjoint trajectory errors decrease at first order, a somewhat better behavior than predicted by the consistency theory. The BDF2 discrete adjoint error is $\mathcal{O}(1)$ and does not decrease with increasing N. The initial time errors (5.2) in Figure 5.2(b), however, have a different behavior. For all the discrete and continuous adjoint methods the initial time errors $E_0(N)$ decrease at the theoretical rates, i.e. second order for AB2 and BDF2 and third order for AB3.

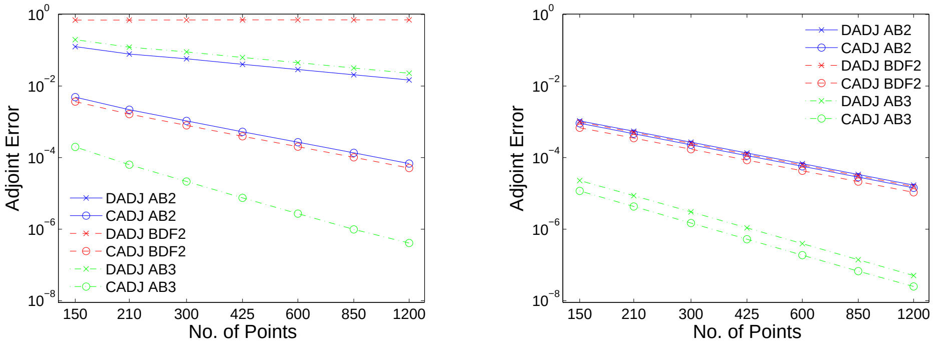
Figure 5.3 shows the work-precision diagrams for fixed step size integration. For continuous adjoint methods both the trajectory and the final time errors decrease



(a) Fixed steps

(b) Variable steps

FIG. 5.1. *Time evolution of E_n errors (5.2) in continuous and discrete adjoint solution errors computed with $N = 25$ time steps.*



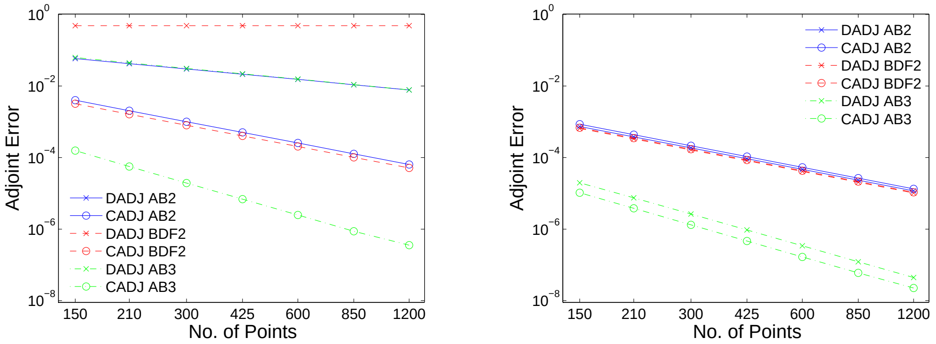
(a) Entire trajectory ($\|E\|(N)$)

(b) Initial time ($E_0(N)$)

FIG. 5.2. *Work-precision diagram for continuous and discrete adjoint solution errors computed with variable time steps. The abscissas represent number of discretization time points N.*

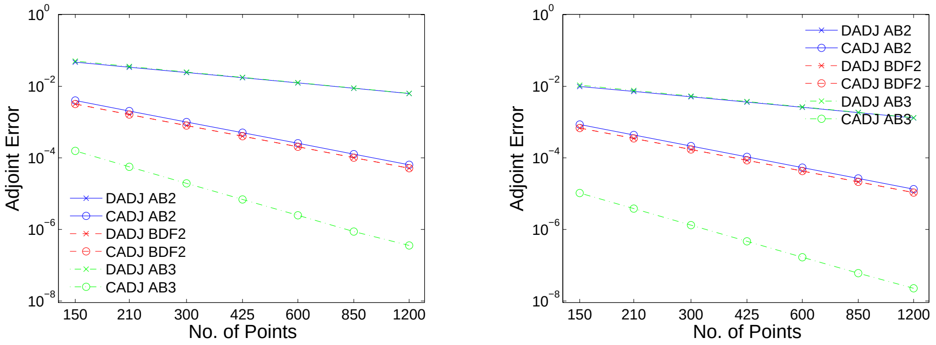
at the theoretical rate. The trajectory errors of the AB2 and AB3 discrete adjoints converge at first order rates, while the initial time errors converge at second and third order rates, respectively. The trajectory error of the BDF2 discrete adjoint does not decrease with the number of steps, while the initial time error converges at second order rate. This behavior seems curious at first since the discrete adjoint BDF2 for constant steps is the BDF2 method at the internal grid points. However, the initialization and termination steps of the discrete adjoint BDF2 are inconsistent with the adjoint ODE.

We next modify the initialization and termination relations in the fixed step discrete adjoint integrations to match those used by the corresponding continuous adjoint integrations. This ensures that each discrete adjoint is used with consistent initial and terminal steps. The resulting work-precision diagrams are shown in Figure 5.4. The discrete AB2 and AB3 adjoints become, effectively, the one-leg methods associated with AB2 and AB3; both the trajectory and the initial time errors decrease at a first order rate. The discrete BDF2 adjoint is effectively the BDF2 applied to the adjoint ODE, and both the trajectory and the initial time errors decrease at a second order rate.



(a) Entire trajectory ($\|E\|(N)$) (b) Initial time ($E_0(N)$)

FIG. 5.3. *Work-precision diagram for continuous and discrete adjoint solution errors computed with fixed time steps. The abscissae represent the number of discretization time points N.*



(a) Entire trajectory ($\|E\|(N)$) (b) Initial time ($E_0(N)$)

FIG. 5.4. *Work-precision diagram for adjoint errors with fixed time steps. The initialization and the termination conditions have been replaced by consistent ones.*

6. Conclusions. In this paper we have derived the discrete adjoints of linear multistep formulas and have analyzed their consistency properties. Discrete adjoints are very popular in optimization and control since they can be constructed automatically by reverse mode automatic differentiation.

In general the discrete LMM adjoints are not consistent with the adjoint ODE along the trajectory when variable time steps are used. If the forward LMM integration is zero-stable then the discrete adjoint process is zero-stable as well. For fixed time steps the discrete adjoint steps are consistent with the adjoint ODE at the internal grid points but not at the initial and terminal points. The initialization and termination steps in the fixed step discrete adjoint process can be changed to obtain consistent schemes.

The discrete adjoints at the initial time, however, converge to the continuous adjoint at a rate equal to the convergence order of the original LMM. This remarkable property is due to the linearity of the method and of its initialization procedure. Therefore the discrete adjoint linear multistep methods provide accurate gradients of a cost functional with respect to a set of fixed model parameters.

Numerical tests on the Arenstorf orbit system confirm the theoretical findings.

The discrete adjoint BDF2 solution is not consistent with the continuous adjoint solution at intermediate integration times, and the numerical error is heavily influenced by the pattern of step size changes. The fixed step BDF2 adjoint is not consistent due to initialization and termination procedures. When these steps are changed the solution converges at second order. The discrete AB2 and AB3 adjoints converge at first order. For all methods the discrete adjoints at the initial time convergence at the theoretical order of the forward methods.

Future work will be devoted to the error analysis of discrete adjoints in the case of stiff systems. ODE solvers automatically adjust the step size during the forward integration to keep an estimate of the local error below a user given threshold. This introduces a dependency of the step size on the forward solution. The effect of such a step control mechanism on the the discrete adjoint method will also be considered in the follow-up work.

Acknowledgments. This work was supported by the National Science Foundation (NSF) through the awards NSF CAREER ACI-0413872, NSF ITR AP&IM 020519, and NSF CCF–0515170, by the National Oceanic and Atmospheric Administration (NOAA) and by the Texas Environmental Research Consortium (TERC). The author thanks Mihai Alexe who helped with running automatic differentiation on some of the codes.

REFERENCES

[1] M.L. Baguer and W. Romisch. *Computing gradients in parametrization-discretization schemes for constrained optimal control problems.* Approximation and Optimization in the Caribbean II. Peter Lang, Frankfurt am Main, 1995.

[2] M.B. Giles. *On the use of Runge-Kutta time-marching and multigrid for the solution of steady adjoint equations.* Technical Report NA00/10, Oxford University Computing Laboratory, 2000.

[3] W. Hager. *Runge-Kutta methods in optimal control and the transformed adjoint system.* Numerische Mathematik, 87(2):247–282, 2000.

[4] E. Hairer, S.P. Norsett, and G. Wanner. *Solving ordinary differential equations I. Nonstiff problems.* Springer-Verlag, Berlin, 1993.

[5] E. Hairer and G. Wanner. *Solving ordinary differential equations II. Stiff and differential-algebraic problems.* Springer-Verlag, Berlin, 1991.

[6] P. Miehe and A. Sandu. *Forward, tangent linear, and adjoint Runge Kutta methods in KPP–2.2 for efficient chemical kinetic simulations.* Technical Report CS-TR-06-27, Computer Science Department, Virginia Polytechnic Institute and State University, 2006.

[7] A. Sandu. *On the properties of Runge-Kutta discrete adjoints.* Lecture Notes in Computer Science, volume LNCS 3994, Part IV, pages 550–557. International Conference on Computational Science, 2006.

[8] A. Sandu, D. Daescu, and G.R. Carmichael. *Direct and adjoint sensitivity analysis of chemical kinetic systems with KPP: I – Theory and software tools.* Atmospheric Environment, 37:5083–5096, 2003.

[9] A. Sandu and L. Zhang. *Discrete second order adjoints in atmospheric chemical transport modeling.* Technical Report CS-TR-07-35, Computer Science Department, Virginia Polytechnic Institute and State University, 2007.

[10] A. Walther. *Automatic differentiation of explicit Runge-Kutta methods for optimal control.* Technical Report WR-06-2004, Technical University Dresden, 2004.