Computational Aspects of Chemical Data Assimilation into Atmospheric Models

Gregory R. Carmichael[1], Dacian N. Daescu[2], Adrian Sandu[3], and Tianfeng Chai[1]

[1] Center for Global and Regional Environmental Research, 204 IATL,The University of Iowa, Iowa City, IA 52242-1297. gcarmich@cgrer.uiowa.edu
Phone: 319-335-3333. Fax: 319-335-3337.

[2] Institute for Mathematics and its Applications, University of Minnesota, 400 Lind Hall 207 Church Street S.E. Minneapolis, MN 55455. daescu@ima.umn.edu

[3] Department of Computer Science, Michigan Technological University, 1400 Townsend Drive, Houghton, MI 49931. asandu@mtu.edu

Abstract. The task of providing an optimal analysis of the state of the atmosphere requires the development of novel computational tools that facilitate an efficient integration of observational data into models. In this paper we discuss some of the computational tools developed for the assimilation of chemical data into atmospheric models. We perform a theoretical analysis of discrete and continuous adjoints for stiff differential equation solvers. Software tools particularly tailored for direct and adjoint sensitivity analysis of chemical systems are presented. The adjoint of the state-of-the-art model STEM-III is discussed, together with ozone assimilation results for a realistic test problem.

Keywords: Data assimilation, Chemical transport models, Adjoint modeling.

1 Introduction

Our ability to anticipate and manage changes in the atmospheric pollutant concentrations relies on an accurate representation of the chemical state of the atmosphere. As our fundamental understanding of atmospheric chemistry has significantly advanced, novel computational tools are needed to integrate observational data and models with the goal of providing an optimal analysis state of the atmosphere. By optimal analysis state we mean an intimate and close integration of modeled and measured quantities, with the two merged together to provide the best estimate, physically consistent, of the evolving chemical state of the atmosphere. The analysis state better defines the spatial and temporal fields of key chemical components in relation to their sources and sinks. This information is critical in designing cost-effective emission control strategies for improved air quality, for the interpretation of observational data such as those obtained during intensive field campaigns, and to the execution of air-quality forecasting.

In this paper we present some of the computational tools developed for chemical data assimilation into atmospheric transport models. The paper is organized

P.M.A. Sloot et al. (Eds.): ICCS 2003, LNCS 2660, pp. 269–278, 2003.

as follows. In Section 2 we review the adjoint sensitivity analysis technique and in Section 3 we discuss the derivation of the adjoint model associated with one step stiff numerical integrators. Software tools that enable an efficient implementation are presented in Section 4. Results from a 3-dimensional assimilation test problem are shown in Section 5, and in Section 6 we draw conclusions and pinpoint future work.

2 Adjoint Sensitivity Analysis

Consider the adjoint sensitivity analysis of stiff nonlinear ordinary differential equations arising in the mathematical formulation of the chemical reactions mechanisms

$$\frac{dy}{dt} = f(t,y)\ , \qquad y\left(t^0\right) = y^0\ , \qquad t^0 \le t \le t^{\mathrm{F}}\ . \tag{1}$$

where $y(t) \in \Re^n$ represents the vector of concentrations of the chemical species in the model. To simplify the presentation, in this paper we consider as parameters the initial state y^0 of the model; it is known that this does not restrict the generality of the formulation. We will refer to the dynamical model (1) as a forward (direct) model; its solution $y = y(t, y^0)$ is uniquely determined once the vector of model parameters y^0 is specified.

Data assimilation applications require the sensitivities (gradient) of a scalar response function $g = g\big(y(t^{\mathrm{F}})\big)$ with respect to the model parameters $y^0_1 \dots y^0_n$. When the number of parameters is large, the adjoint method provides an efficient alternative to the direct decoupled method [7] for evaluating these sensitivities. Mathematical foundations of the adjoint sensitivity for nonlinear dynamical systems are presented by Cacuci [2, 3] and Marchuk et al. [13, 14]. Specific applications of adjoint modeling to atmospheric chemical data assimilation are described by Menut et al. [12], Vautard et al. [16], Wang et al. [20], and Elbern et. al. [8].

In the adjoint sensitivity analysis one distinguishes between the continuous and the discrete adjoint modeling, see Sirkes [19].

2.1 Continuous Adjoint Sensitivity

The continuous adjoint model is obtained from the forward model using the linearization technique [2]. Define the "adjoint variable" $\lambda(t) \in \Re^n$ as the solution of the adjoint problem

$$\frac{d\lambda}{dt} = -J^T(t,y)\lambda\ , \quad \lambda(t^{\mathrm{F}}) = \nabla_y g(y(t^{\mathrm{F}})) \tag{2}$$

The continuous method consists in applying a numerical discretization scheme to integrate (2) backwards in time; the sensitivity values are

$$\nabla_{y^0} g = \lambda(t^0)\ . \tag{3}$$

2.2 Discrete Adjoint Sensitivity

The discretization of the system (1) with a selected numerical method results in the discrete forward model, which obtains a numerical approximation $y^N \approx y(t^{\mathrm{F}})$ through a sequence of N intermediate states

$$y^{i+1} = F^i(y^i) \ , \quad i = 0, \ldots, N-1 \ , \tag{4}$$

where F^i represents a one-step numerical integration formula which advances the solution from t^i to t^{i+1}. To obtain the adjoint sensitivity of the numerical solution we evaluate the adjoint variables at $t^i, i = N-1, \ldots, 1, 0$ using the recursive relations

$$\lambda^N = \nabla_y g(y^N) \ , \quad \lambda^i = F^i_y(y^i, p^i)^T \lambda^{i+1} \ . \tag{5}$$

The discrete functional sensitivities are

$$\nabla_{y^0} g(y^N) = \lambda^0 \ . \tag{6}$$

3 Stiff Numerical Methods and Their Adjoints

Here we consider one-step numerical methods for the integration of stiff ordinary differential equations and discuss the corresponding discrete adjoints and their consistency with the continuous adjoint equation. This analysis is relevant for atmospheric data assimilation in the context of modeling stiff chemical processes.

3.1 Runge-Kutta Methods

Consider the s-stage Runge Kutta method [10, 11]

$$y_{n+1} = y_n + h \sum_{j=1}^{s} b_j f(z_j) \ , \quad z_i = y_n + h \sum_{j=1}^{s} a_{ij} f(z_j) \ . \tag{7}$$

The Discrete Adjoint. In what follows we will use the notation

$$L_j = h\, J^T(t_n + c_j h, z_j) \approx J(t, y(t))|_{t=t_n+c_j h} \ , \quad j = 1 \cdots s \ ,$$

The discrete adjoint of the Runge-Kutta scheme reads [9]

$$\lambda_n = \lambda_{n+1} + \sum_{j=1}^{s} \theta_j \ , \quad \theta_i = L_i \left[b_i \lambda_{n+1} + \sum_{j=1}^{s} a_{j,i} \theta_j \right] \ ,. \tag{8}$$

The (backwards) propagation of the discrete adjoints defines the discrete adjoint transfer function

$$\lambda_n = \left(\frac{\partial y_{n+1}}{\partial y_n} \right)^T \lambda_{n+1} = R_D \ \lambda_{n+1} \ . \tag{9}$$

The Continuous Adjoint. Consider the adjoint equation (2) and discretize it directly using a Runge-Kutta method (7) with coefficients $\tilde{a}_{ij}$, $\tilde{b}_i$, $\tilde{c}_j$. We denote

$$M_i = \, h\, J^T(t, y(t))\big|_{t=t_{n+1}-\tilde{c}_i h}$$

The method is applied backwards in time (from t^{n+1} to t^n)

$$\lambda_n = \lambda_{n+1} + \sum_{j=1}^{s} \tilde{b}_j M_j \cdot w_j \ , \quad w_i = \lambda_{n+1} + \sum_{j=1}^{s} \tilde{a}_{ij} M_j \cdot w_j \ . \tag{10}$$

This (backwards) propagation of the discretized adjoints defines the continuous adjoint transfer function

$$\lambda_n = R_C \ \lambda_{n+1} \tag{11}$$

Comparison of the two approaches. Hager [9] noticed that, if $L_i = M_i$ and $b_i \neq 0$, the discrete adjoint is equivalent to a discretization of the continuous adjoint equation by a Runge Kutta scheme with

$$\tilde{a}_{ij} = \frac{a_{s+1-j,s+1-i}\, b_{s+1-j}}{b_{s+1-i}} \ , \quad \tilde{b}_i = b_{s+1-i} \ , \quad \tilde{c}_i = 1 - c_{s+1-i} \ . \tag{12}$$

Order Conditions.

Theorem 1. *Consider a Runge-Kutta method (7) consistent of order p and its discrete adjoint (9). The relation (9) is a numerical discretization of the continuous adjoint equation (2), which is consistent of order p.*

Proof. The proof is given in [18]. Based on the order condition theory for Runge Kutta methods [10, Section II.2] one expresses the entries of the discrete and the continuous transfer functions in terms of elementary differentials, and establishes that the order conditions for the discrete adjoint are the same as for the original Runge Kutta method.

Control Problems. Control Problems use a formulation where the function f depends on the adjoint $f(y, \lambda)$ (due to its dependency of the control variable). In [18] we analyze the consistency of the discrete adjoint (9) with (2) and generalize the findings of Hager [9].

If all $b_i \neq 0$ we can rewrite the discrete adjoint relation as (see [18])

$$\lambda_{n+1} = \lambda_n + h \sum_{j=1}^{s} b_j \ell_j \ , \quad \ell_i = -J^T(z_i) \ \left[\lambda_n + h \sum_{j=1}^{s} \bar{a}_{i,j} \ell_j \right] \ . \tag{13}$$

where $\bar{a}_{i,j} = b_j - (a_{j,i} b_j)/b_i$.

Consider now the forward model (1) in tandem with its adjoint (2) as a partitioned ordinary differential equation and solve it using the partitioned RK method given by (7) and (13).

To obtain the order of the partitioned method we follow the P-tree theory of [10, p. 309]. We associate meagre vertices with y and fat vertices with λ. For control problems we recover the extra conditions of Hager [9]. For example, the extra condition for order 3 corresponds to the following P-tree • — — ○ — — •; this P-tree has a vanishing differential if f does not depend on the adjoint, $f = f(y)$. Obtaining higher order conditions of consistency is a tedious but straightforward process.

Singular Perturbation Analysis. In [18] we analyze the behavior of adjoints in the stiff case by considering the singularly perturbed problem

$$y' = f(y,z)\ , \quad \epsilon z' = h(y,z)\ , \quad t^0 \le t \le t^F \tag{14}$$

with the sub-Jacobian h_z assumed nonsingular. This model allows us to distinguish between the derivatives of the functional $g(y(t^F), z(t^F))$ with respect to non-stiff and stiff variables

$$\lambda(t) = \frac{\partial g(y(t^F), z(t^F))}{\partial y(t)}\ , \quad \mu(t) = \frac{\partial g(y(t^F), z(t^F))}{\partial z(t)}\ .$$

We consider ϵ-expansions of the adjoint variables $\lambda = \lambda^0 + \epsilon\lambda^1 + \ldots$ and $\mu = \mu^0 + \epsilon\mu^1 + \ldots$ The first terms of these expansions satisfy

$$(\lambda^0)' = \left(-f_y^T + h_y^T h_z^{-T} f_z^T\right)\ \lambda^0 \quad \text{and} \quad \mu^0 = 0\ . \tag{15}$$

Consider a Runge Kutta method with invertible coefficient matrix A and with the stability function satisfying $R(\infty) = 0$. In [18] it is shown that, if the cost function depends only on the non-stiff variable y (i.e. we initialize $\mu_N^0 = 0$) and $R(\infty) = 0$ then $\mu_n^0 = 0$ for all n. In this case the discrete adjoint is a consistent discretization of the reduced equation (15), and the values of λ are solved with the same accuracy as the original method, within $\mathcal{O}(\epsilon)$. A similar conclusion holds for continuous Runge Kutta adjoints.

3.2 Rosenbrock Methods

Consider an autonomous Rosenbrock method in the implementation-friendly form

$$y_{n+1} = y_n + \sum_{j=1}^{s} m_j k_j\ , \quad Y_i = y_n + \sum_{j=1}^{i-1} a_{i,j} k_j\ , \tag{16}$$

$$\left(\frac{1}{h\gamma} - J(y_n)\right)\ k_i = f\left(Y_i\right) + \sum_{j=1}^{i-1} \frac{c_{i,j}}{h} k_j\ , \quad i = 1, \cdots, s\ . \tag{17}$$

The Discrete Adjoint In [18] we show that the discrete adjoint of the Rosenbrock method (17) reads

$$\left(\frac{1}{h\gamma} - J^T(y_n)\right) u_i = m_i \lambda_{n+1} + \sum_{j=i+1}^{s} \left(a_{j,i} v_j + \frac{c_{j,i}}{h} u_j\right) \,, \tag{18}$$

$$v_i = J^T(Y_i) u_i \,, \quad i = s, s-1, \cdots, 1 \,, \tag{19}$$

$$\lambda_n = \lambda_{n+1} + \sum_{i=1}^{s} \left(H(y_n) \times k_i\right)^T \cdot u_i + \sum_{i=1}^{s} v_i \tag{20}$$

Here J denotes the Jacobian and H (a 3-tensor) is the Hessian of the derivative function f. The formulation can be easily extended to non-autonomous systems.

The Continuous Adjoint The continuous adjoint is obtained by solving (2) with the Rosenbrock method (17).

4 Software Tools for Adjoint Sensitivity of Chemical Kinetic Systems

In this section we present the Kinetic PreProcessor KPP software tools that are useful in derivative computations. A detailed discussion of the basic KPP capabilities can be found in our previous work [6]. Here we focus on the new features introduced in the release 1.2 [17, 5] that allow a quick and efficient implementation of the code for sensitivity analysis of chemical kinetic systems.

KPP builds Fortran or C simulation code for chemical systems with chemical concentrations changing in time according to the law of mass action kinetics. KPP generates the following building blocks:

1. *FunVar*: the time derivative of concentrations;
2. *JacVar, JacVar_SP*: Jacobian of FunVar in full or in sparse format;
3. *KppDecomp*: sparse LU decomposition for the Jacobian;
4. *KppSolve, KppSolveTR*: solve sparse system with the Jacobian matrix and its transpose;
5. *JacVar_SP_Vec, JacVarTR_SP_Vec*: sparse Jacobian (transposed or not) times vector;
6. The stoichiometric matrix STOICM;
7. *ReactantProd*: vector of reaction rates;
8. *JacVarReactantProd*: the Jacobian of the above;
9. *dFunVar_dRcoeff*: derivatives of FunVar with respect to reaction coefficients (in sparse format);
10. *dJacVar_dRcoeff*: derivatives of JacVar with respect to reaction coefficients times user vector;
11. *HessVar*: the Hessian of FunVar; this 3-tensor is represented in sparse format;

12. *HessVar_Vec, HessVarTR_Vec*: Hessian (or its transpose) times user vectors; same as the derivative of Jacobian (transposed) vector product times vector.

In [17, 5] we show how the KPP building blocks can be used to implement very efficiently code for direct and adjoint sensitivity analysis of chemical systems.

5 Numerical Results with Adjoint STEM-III

We now present assimilation results for a problem being analyzed in support of large field experiments conducted in East Asia (i.e. TraceP and AceAsia experiments). The simulated region is East Asia, the simulated interval is 6 hours starting at 0 GMT on March 1st, 2001. The meteorological fields are given by a dynamic meteorological model (RAMS), and the initial fields and boundary conditions correspond to Trace-P data campaign. The grid is $90 \times 60 \times 16$ points and has a horizontal resolution of $80Km \times 80Km$.

The adjoint STEM-III code has the following characteristics:

- The transport is computed using an upwind third order finite difference scheme for advection, and a second order centered scheme for diffusion, with Crank-Nicholson time integration. The transport operator is linear and the implementation of its adjoint straightforward;
- The gas phase chemical mechanism is SAPRC-99 [4] which considers the gas-phase atmospheric reactions of volatile organic (VOCs) and nitrogen oxides (NOx) in urban and regional settings. The forward time integration is done with the Rosenbrock numerical integrator Ros-2; the continuous adjoint model uses Ros-2 on the same sequence of steps as the forward chemical integration. Both the forward and the adjoint models are implemented using KPP.
- The forward and adjoint models are parallel and were run on a cluster of Linux workstations. Parallelization is based on our library PAQMSG [15].
- The checkpoints store the concentration fields every 15 minutes, i.e. for every operator split step. The checkpoints are distributed as seen in Figure 2, which means that each node stores local information on the local disk. This increases the storage capabilities and decreases the input/output overhead when the parallel computation runs on a cluster of workstations.
- On 16 nodes (Pentium, 2GHz, 1GB RAM) the cpu time for a forward run is about 2 minutes per hour of simulation; and the cpu time for a forward-backward run is about 4.5 minutes per hour of simulation.

The data assimilation procedure is set using the twin experiments method as follows:

Reference run: we start a model run at $t^s = 0{:}00GMT$ with the standard concentrations of all variable chemical species.

Initial guess run: the experiment is repeated with ozone initial concentrations increased by 20%.

Observations and assimilation window: We consider a 6 hours assimilation window. The observations for O_3 concentrations on all grid points are provided only at the end of the assimilation window $t^0 + 6h$.

Parameters: the control parameters are the ozone concentrations at t^0 .

Cost functional: Information to the assimilation process is provided only by the "observations" at the final time. To achieve a better scaling and to eliminate the positivity constraint the logarithmic form is used

$$\mathcal{J} = \frac{1}{2} \sum_{\text{gridpoints}} [(\ln O_3(t^F) - \ln O_3^{\text{obs}}(t^F)]^2 \ . \tag{21}$$

Optimization algorithm: Quasi-Newton limited memory L-BFGS [1]. The optimization proceeds until the cost functional is reduced to 0.001 of its initial value.

The results of the data assimilation are presented in Figure 1. We notice a sharp decrease in the cost function value during the first optimization iterations, followed by a slower decrease rate. The ozone concentration at gridpoint (40,30,1) is recovered within 6 accurate digits. The root mean square difference between the assimilated and the reference fields decreases to about 1e-3. Larger assimilation errors are noted near the top and bottom domain boundaries, which requires further algorithmic developments.

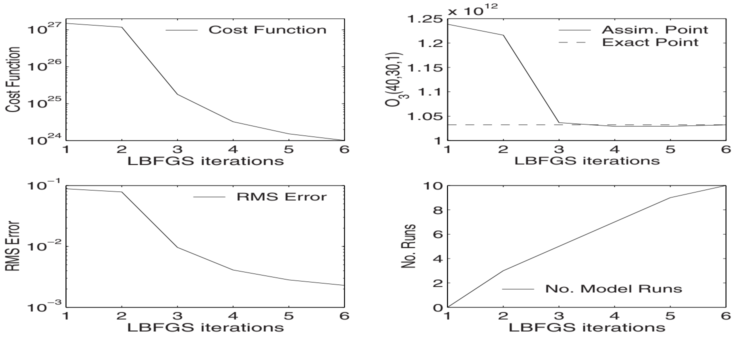

Fig. 1. Data assimilation results for ozone, for a 6h run.

6 Conclusions and Future Work

In this paper we presented some of the computational tools developed for chemical data assimilation into atmospheric transport models. The focus is on developing efficient adjoints for the stiff ordinary differential equations arising in the simulation of chemistry.

We recall the analysis of discrete Runge Kutta adjoints performed by Hager [9] and develop the discrete adjoints for Rosenbrock methods. We establish that

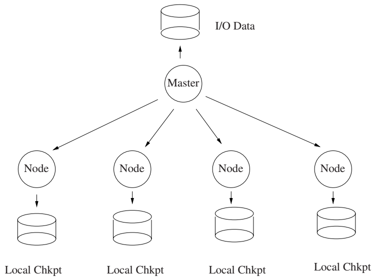

Fig. 2. The parallel adjoint STEM implements a distributed checkpointing scheme.

Runge Kutta discrete adjoints are consistent with the continuous adjoint equation with the same order as the underlying method. For control problems we propose an order analysis in the framework of partitioned Runge Kutta methods; this analysis enables an extension of the original analysis [9]. A singular perturbation analysis enables us to establish that L-stability is beneficial not only for the integration of the stiff forward model, but also for the backwards integration with both the discrete and the continuous adjoints.

The Kinetic PreProcessor KPP developed by the authors is a symbolic engine that translates a given chemical mechanism into Fortran or C kinetic simulation code. A comprehensive set of software tools for direct and adjoint sensitivity analysis were developed and implemented in the new release of the Kinetic PreProcessor (KPP-1.2). They include code generation for the Hessian and several other derivatives, as well as dedicated sparse linear algebra routines.

An adjoint of the state-of-the-science chemical transport model STEM-III was developed; it uses an efficient distributed checkpointing scheme; the chemical subsystem is generated with KPP. Results are shown for a 6 hours test case based on the TraceP scenario, where the perturbed initial concentrations of ozone are recovered from the known ozone concentrations at the end of the simulation interval.

Future work will focus on continuing the theoretical analysis of discrete adjoints for stiff solvers; on using the developed computational infrastructure to run more complex tests and to assimilate real measurements data; and on continuing to improve the computational infrastructure for data assimilation.

Acknowledgements.

The authors thank the National Science Foundation for supporting this work through the award NSF ITR AP&IM 0205198.

References

1. R. Byrd, P. Lu, and J. Nocedal. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Stat. Comput.*, 16(5):1190–1208, 1995.

2. D. G. Cacuci. Sensitivity theory for nonlinear systems. I. Nonlinear functional analysis approach. *J. Math. Phys.*, 22:2794–2802, 1981.
3. D. G. Cacuci. Sensitivity theory for nonlinear systems. II. Extensions to additional classes of responses. *J. Math. Phys.*, 22:2803–2812, 1981.
4. W.P.L. Carter. Implementation of the SAPRC-99 Chemical Mechanism into the Models-3 Framework. Technical report, Report to the United States Environmental Protection Agency, January 2000.
5. D. Daescu, A. Sandu, and G.R. Carmichael. Direct and Adjoint Sensitivity Analysis of Chemical Kinetic Systems with KPP: II – Numerical Validation and Applications. *submitted to Atmospheric Environment*, 2002.
6. V. Damian, A. Sandu, M. Damian, F. Potra, and G.R. Carmichael. The kinetic preprocessor kpp – a software environment for solving chemical kinetics. *(Computers and Chemical Engineering)*, 26:1567–1579, 2002.
7. A. M. Dunker. The decoupled direct method for calculating sensitivity coefficients in chemical kinetics. *Journal of Chemical Physics*, 81:2385, 1984.
8. H. Elbern and H. Schmidt. Ozone episode analysis by 4D-Var chemistry data assimilation. *Journal of Geophysical Research*, 106(D4):3569–3590, 2001.
9. William W. Hager. Runge-Kutta methods in optimal control and the transformed adjoint system. *Numerische Mathematik*, 87(2):247–282, 2000.
10. E. Hairer, S.P. Norsett, and G. Wanner. *Solving Ordinary Differential Equations I. Nonstiff Problems.* Springer-Verlag, Berlin, 1993.
11. E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems.* Springer-Verlag, Berlin, 1991.
12. Menut L., Vautard R., Beekmann M., , and Honor C. Sensitivity of photochemical pollution using the adjoint of a simplified chemistry-transport model. *Journal of Geophysical Research - Atmospheres*, 105-D12(15):15,379–15,402, 2000.
13. G.I. Marchuk. *Adjoint Equations and Analysis of Complex Systems.* Kluwer Academic Publishers, 1995.
14. G.I. Marchuk, Agoshkov, and P.V. I.V., Shutyaev. *Adjoint Equations and Perturbation Algorithms in Nonlinear Problems.* CRC Press, 1996.
15. P. Miehe, A. Sandu, G.R. Carmichael, Y. Tang, and D. Daescu. A communication library for the parallelization of air quality models on structured grids. *(Atmospheric Environment)*, 36:3917–3930, 2002.
16. Vautard R., M.Beekmann, and L. Menut. Applications of adjoint modelling in atmospheric chemistry: sensitivity and inverse modelling. *Environmental Modeling and Sofware*, 15:703–709, 2000.
17. A. Sandu, D. Daescu, and G.R. Carmichael. Direct and Adjoint Sensitivity Analysis of Chemical Kinetic Systems with KPP: I – Theory and Software Tools. *submitted to Atmospheric Environment*, 2002.
18. A. Sandu, D. daescu, and G.R. Carmichael. Discrete Adjoint for Stiff ODE Solvers. *In Preparation*, 2003.
19. Z. Sirkes and E. Tziperman. Finite difference of adjoint or adjoint of finite difference? *Mon. Weather Rev.*, 49:5–40, 1997.
20. K.Y. Wang, D.J. Lary, Shallcross, D.E., Hall aS.M., and Pyle J.A. A review on the use of the adjoint method in four-dimensional atmospheric-chemistry data assimilation. *Q.J.R. Meteorol. Soc.*, 127(576 (Part B)):2181–2204, 2001.