

Tangent linear and adjoint biogeochemical models

Ralf Giering

*Jet Propulsion Laboratory
California Institute of Technology, California*

Adjoint models are powerful tools for inverse modeling. They are increasingly being used in meteorology and oceanography for sensitivity studies, data assimilation, and parameter estimation. Covering the range from simple box models to sophisticated General Circulation Models, they efficiently compute the sensitivity of a few model output variables with respect to arbitrarily many input variables. To the contrary, tangent linear models efficiently compute the model output perturbation resulting from an initial input perturbation. Mathematically, both models evaluate the first derivative or Jacobian matrix of the mapping defined by the original model. Efficiency is an important issue for sophisticated models and in practice often determine whether a problem is solvable or not. We discuss here the advantages of tangent linear and adjoint models, as well as when to use either of them. The construction of adjoint and tangent linear models by hand is tedious and error prone. Computational Differentiation reduces this work substantially. The basics of Computational Differentiation, its advantages and limitations are presented. Three different kinds of sensitivities are shown: the sensitivity of a passive tracer concentration in the ocean to previous concentrations, the sensitivity of atmospheric CO₂ at Mauna Loa, Hawaii, to surface fluxes, and the sensitivity of the North Atlantic meridional heat flux to subsurface salinity. They are discussed in terms of the chains of cause and effect. For biogeochemical models of passive tracers sensitivities can be explained by advective and diffusive processes. However, for dynamically active tracers, the sensitivities are modified by an additional path of influence.

1. INTRODUCTION

Adjoint model (ADM) and tangent linear models (TLM) are increasingly being developed and used in meteorology and physical oceanography. Typical applications are data assimilation, parameter estimation, sensitivity analysis and determination of singular vectors (*Errico, 1997*).

In canonical form, a model is a mapping of input variables (independent variables) onto output variables (dependent variables). If this mapping is differentiable, its first derivative is the Jacobian matrix. A TLM is a program to compute the action of the Jacobian matrix on a vector. Here, the vector is mostly a perturbation of the initial condition or the boundary forcing. The TLM is linear and it depends on the model trajectory at which the linearization took place. The TLM simulates the development of perturbations with time

and can be used to analyze the impact of small disturbances. For instance, consider a TLM of the advection of a passive tracer by horizontal currents. If the tracer concentration at one point is changed, this anomaly is transported downstream and broadened by diffusion (Fig. 1).

In contrast, an ADM is a program to compute the action of the transposed Jacobian matrix on a vector. It is adjoint to the tangent linear model! The ADM is linear and depends on the model trajectory at which the linearization took place in the same manner as the TLM. The vector is an arbitrary sensitivity and the ADM simulates the development of sensitivities backward. The ADM can be used to analyze the origin of any anomaly. As shown in Figure 2, a difference at one location can be caused by propagation of an anomaly from upstream. Thereby, due to the effect of diffusion, the possible origin of the anomaly is located in a broader area.

There are some limitations of TLMs and ADMs. If, at some locations, the underlying function is non differentiable, sensitivities cannot be determined or they might be misleading. In ocean and atmospheric models sub grid processes are often formulated in a non differentiable way. If, e.g., convective adjustment is used for unstable stratification in an ocean model the sensitivities do not represent a change in the convection pattern. Thus, a revised formulation of the parametrization of sub grid processes might be necessary (*Xu, 1996a,b; Zou, 1997*).

TLM's and ADM's are based on the linear approximation. For non linear models the sensitivities are only valid at a certain point in phase space. For highly non-linear or chaotic models the computed sensitivities might change rapidly with a varying point of linearization. In some extremes, this could make these sensitivities totally useless.

Direct coding of ADMs and TLMs of sophisticated model is extremely time consuming and subject to errors. Hence, automatic generation of ADMs and TLMs represents a distinct advantage. In computational science this is known as reverse and forward mode of Automatic or Computational Differentiation. A tool of Computational Differentiation is the Tangent linear and Adjoint Model Compiler (TAMC, R. Giering, unpublished manual, 1997). This tool has been used to generate several ADMs and TLMs. The performance of the generated code is comparable to hand written models (*Giering and Kaminski, 1998b*).

Here we focus on the application of adjoint models for sensitivity studies. In three examples of biogeochemical

models adjoint sensitivities of different output variables with respect to different input variables are discussed. For passive tracers in the ocean and atmosphere sensitivities can be explained by horizontal advection, vertical advection, and diffusive processes. In contrast, an active tracer influences the model dynamics and has an additional influence on the system.

The outline of the paper is as follows. After presenting the mathematical background in section 2 the applications of TLMs and ADMs are described in section 3. Section 4 explains the different methods of tangent linear and adjoint code constructions and gives a short introduction into Computational Differentiation. Fields of sensitivities for three different models are presented and explained in section 5.

2. MATHEMATICAL BACKGROUND

In the following the mathematical background of tangent linear and adjoint models is described by the variational formalism. In the literature other descriptions can be found, which are commonly based on the Lagrange function (e.g. *Thacker and Long, 1988; Schröter, 1989*). By using the variational formalism, the connection between adjoint models and the adjoint operator of Linear Algebra is more obvious. Here we introduce adjoint models in terms of data assimilation for pedagogical reasons.

Consider a numerical model describing a dynamical system. Let $\mathbf{y}^o \in \mathbf{R}^m$ ($m \in \mathbf{N}$) be a set of observations and $\mathbf{y} \in \mathbf{R}^m$ the corresponding model values. The misfit between model and observations is usually quantified by a quadratic cost function

$$J := \frac{1}{2} (\mathbf{y} - \mathbf{y}^o, \mathbf{y} - \mathbf{y}^o) \quad (1)$$

by the choice of an appropriate inner product (\cdot, \cdot) . This implies that least-squares-fitting is intended: The smaller J is the better the model fits the data. How can the model be manipulated in order to obtain an optimal fit between observations and corresponding model values?

In order to manipulate the model, we specify a set of $n \in \mathbf{N}$ parameters \mathbf{x} , which, in the following, will be called control variables. The dependence of \mathbf{y} on \mathbf{x} within the model is given by the mapping

$$\begin{aligned} \mathcal{F} : \mathbf{R}^n &\rightarrow \mathbf{R}^m \\ \mathbf{x} &\mapsto \mathbf{y} . \end{aligned} \quad (2)$$

This mapping usually consists of the time integration of the model and a mapping of the state vector to the

observed values. Thus, J can be expressed in terms of \mathbf{x} by

$$\begin{aligned} J : \mathbb{R}^n &\rightarrow \mathbb{R} \\ \mathbf{x} &\mapsto \frac{1}{2} (\mathcal{F}(\mathbf{x}) - \mathbf{y}^o, \mathcal{F}(\mathbf{x}) - \mathbf{y}^o) . \end{aligned} \quad (3)$$

The problem we want to solve is to determine the set of control variables \mathbf{x} that minimizes J . Efficient minimization algorithms make use of the gradient $\nabla_{\mathbf{x}} J(\mathbf{x}_i)$ of J with respect to the control variables at a given point \mathbf{x}_i . To first order we write the Taylor expansion of J :

$$J(\mathbf{x}) = J(\mathbf{x}_i) + (\nabla_{\mathbf{x}} J(\mathbf{x}_i), \mathbf{x} - \mathbf{x}_i) + o(|\mathbf{x} - \mathbf{x}_i|) \quad (4)$$

or, equivalent, as variation

$$\delta J = (\nabla_{\mathbf{x}} J(\mathbf{x}_i), \delta \mathbf{x}) . \quad (5)$$

In the following we will use this shorthand notation whenever linear approximations are involved. Suppose \mathcal{F} is sufficiently regular, then for each control vector \mathbf{x}_i , a variation of \mathbf{y} can be approximated to first order by

$$\delta \mathbf{y} = \mathbf{A}(\mathbf{x}_i) \delta \mathbf{x} , \quad (6)$$

where $\mathbf{A}(\mathbf{x}_i)$ denotes the Jacobian of \mathcal{F} at \mathbf{x}_i .

Due to the symmetry of the inner product and the product rule the differentiation of (3) yields

$$\begin{aligned} \delta J &= \frac{1}{2} (\mathbf{A}(\mathbf{x}_i) \delta \mathbf{x}, \mathcal{F}(\mathbf{x}_i) - \mathbf{y}^o) \\ &+ \frac{1}{2} (\mathcal{F}(\mathbf{x}_i) - \mathbf{y}^o, \mathbf{A}(\mathbf{x}_i) \delta \mathbf{x}) \\ &= (\mathcal{F}(\mathbf{x}_i) - \mathbf{y}^o, \mathbf{A}(\mathbf{x}_i) \delta \mathbf{x}) . \end{aligned} \quad (7)$$

Using the definition of the adjoint operator \mathbf{A}^* :

$$(\mathbf{v}, \mathbf{A} \mathbf{w}) = (\mathbf{A}^* \mathbf{v}, \mathbf{w}) , \quad (8)$$

we obtain

$$\delta J = (\mathbf{A}^*(\mathbf{x}_i)(\mathcal{F}(\mathbf{x}_i) - \mathbf{y}^o), \delta \mathbf{x}) . \quad (9)$$

Therefore, using to the definition of the gradient (5), the gradient of the cost function with respect to the control variables is

$$\nabla_{\mathbf{x}} J(\mathbf{x}_i) = \mathbf{A}^*(\mathbf{x}_i) (\mathcal{F}(\mathbf{x}_i) - \mathbf{y}^o) . \quad (10)$$

The linear operator $\mathbf{A}(\mathbf{x}_i)$ represents the tangent linear model. Its adjoint $\mathbf{A}^*(\mathbf{x}_i)$, which is linear as well, represents the adjoint model. Both operators depend on the point \mathbf{x}_i in phase space, at which the linearization took place. According to (10), the misfit $[\mathcal{F}(\mathbf{x}_i) - \mathbf{y}^o]$ represents the forcing of the adjoint model.

3. APPLICATIONS

3.1. Sensitivity analysis

3.1.1. Forward sensitivity. A standard procedure for studying the impact of specific model parameters or variables on the model trajectory or on derived quantities is to disturb this variable and compare the model response to a 'control run' which was undisturbed. This is a finite difference approximation to the exact sensitivity and is commonly known as the Green's function approach. In contrast, the TLM provides the exact sensitivity (in most cases almost up to machine precision), requires comparable memory resources, but often needs less run time. The integration of the TLM (operator \mathbf{A} in (6)) gives the development δy of an initial perturbation $\delta \mathbf{x}$.

$$\mathbf{A} \delta \mathbf{x} = \delta \mathbf{y} \quad (11)$$

The perturbation vector $\delta \mathbf{x}$ might consist of only one variable or of any linear combination of variables that the model trajectory depends on. The final perturbation vector $\delta \mathbf{y}$ can be the perturbation of the model state or of a number of derived quantities. The TLM computes any linear combination of rows of the Jacobian matrix.

3.1.2. Reverse or backward sensitivity. To answer questions such as, where does a specific anomaly come from or to what is a particular feature most sensitive, a reverse or backward sensitivity is required. Forward sensitivities can hardly answer this efficiently. The ADM integrates sensitivities of a specific feature from the effect to the cause. For time evolving models this means backward in time. The possible cause $\delta^* \mathbf{x}^*$ is the result of the action of the adjoint operator (\mathbf{A}^* in (10)) on the effect $\delta^* \mathbf{y}^*$.

$$\mathbf{A}^* \delta^* \mathbf{y}^* = \delta^* \mathbf{x}^* \quad (12)$$

The ADM does *not* model physical quantities, e.g., concentrations of tracers; instead it models the sensitivities of a property to these quantities. Any linear combination of columns of the Jacobian matrix can be determined by the ADM.

Adjoint sensitivity analysis is well established in meteorology (e.g. *Hall et al.*, 1982; *Errico and Vukicevic*, 1992; *Rabier et al.*, 1992; *Zou et al.*, 1993; *Langland et al.*, 1995). *Kaminski et al.* (1996) determined the sensitivity of the seasonal cycle of atmospheric CO₂ at monitoring stations to the seasonal cycle of surface exchange fluxes. *Oldenborgh et al.* (1999) found adjoint Kelvin and Rossby waves in an adjoint pacific ocean model, carrying sensitivities in the opposite direction to their physical counterparts. The sensitivity of the

North Atlantic annual mean meridional heat flux to initial temperature and salinity was discussed by Maroztke (J. Maroztke et al., submitted manuscript, 1999).

3.2. Data assimilation

Within variational data assimilation, a cost function J

$$J(\mathbf{x}) = (\mathbf{y} - \mathbf{y}^o)^t \mathbf{W}(\mathbf{y} - \mathbf{y}^o) \quad (13)$$

which quantifies the misfit between model and data. is being minimized by varying control variables \mathbf{x} . The misfit is weighted by a matrix \mathbf{W} . Under the assumption of Gaussian error distribution of all contributing errors and when \mathbf{W} is the sum of the associated error covariance matrices, $\min(J(\mathbf{x}))$ corresponds to the maximum likelihood solution. The most important errors are the measurement error, the representation error, and the model error. The representation error is a result of the different spatial and temporal scales resolved by the model and the data. The control variables can be the initial conditions or the boundary conditions. Among the many methods of optimization, gradient methods are the most efficient for differentiable functions. They require computation of the gradient of the cost function with respect to the control variables at each iteration. This gradient is the sensitivity of the cost function with respect to the control variables. It is most efficiently computed by the ADM. Thus, the method is also known as adjoint data assimilation.

This method was first applied in meteorology to simplified models (*Lewis and Derber, 1985*) and is now implemented for weather prediction (e.g. *Courtier et al., 1994*).

Tziperman et al. (1992b) applied the adjoint method to assimilate hydrographic data into an Atlantic ocean model. Several data sets have also been assimilated into a primitive equation global ocean model by varying initial conditions and boundary forcing (*Stammer et al., 1997*).

3.3. Parameter estimation

The estimation of parameters in the underlying equations of a model is very similar to data assimilation. Here the control variables are some of these parameters \mathbf{p} and a cost function $J(\mathbf{p})$ is minimized. Again, the ADM provides the gradient $\nabla_{\mathbf{p}} J$ of the cost function with respect to the parameters. Usually, the number of parameters determined are of the order 10-100. This allows use of more memory intensive optimization algorithm, for example the Newton algorithm. *Navon (1997)* recently reviewed the state of the art in parameter estimation.

3.4. Singular vectors or most unstable modes

In order to forecast the time development of a system, it is useful to know which initial perturbations amplify most rapidly (e.g. *Webster and Hopkins, 1994; Vukicevic, 1998*). If norm $\|\cdot\|_a$ is the measure of an initial perturbation and norm $\|\cdot\|_b$ that of the final perturbation then we need to determine the maximum of:

$$\frac{\|\delta\mathbf{y}\|_b}{\|\delta\mathbf{x}\|_a} \quad (14)$$

Assuming that $\delta\mathbf{x}$ has fixed norm and that $\|\cdot\|_b$ is defined by an appropriate scalar product:

$$\|\mathbf{x}\|_b := (\mathbf{x}, \mathbf{x}) \quad (15)$$

we need to maximize

$$(\delta\mathbf{y}, \delta\mathbf{y}) = (\mathbf{A} \delta\mathbf{x}, \mathbf{A} \delta\mathbf{x}) \quad (16)$$

$$= (\delta\mathbf{x}, \mathbf{A}^* \mathbf{A} \delta\mathbf{x}) \quad (17)$$

by varying only the direction of $\delta\mathbf{x}$. This means that one has to find the largest eigenvalues λ_i and the corresponding eigenvectors \mathbf{v}_i of $\mathbf{A}^* \mathbf{A}$ satisfying:

$$\mathbf{A}^* \mathbf{A} \mathbf{v}_l = \lambda_l \mathbf{v}_l . \quad (18)$$

Thus, a perturbation $\delta\mathbf{x}$ of fixed norm implies the largest possible perturbation $\delta\mathbf{y}$ if it is in the direction associated with the dominant eigenvector of the operator $\mathbf{A}^* \mathbf{A}$ (adjoint times tangent linear operator).

The dominant eigenvectors are called singular vectors or the most unstable modes. They are the solution to a generalized stability problem (*Farrel and Ioannou, 1996a,b*).

3.5. Posterior error estimates

The results of adjoint data assimilation and parameter estimation, the optimal control variables, have uncertainties. These uncertainties are proportional to the curvature of the cost function at its minimum: Strong curvature implies smaller error. Estimates of the errors of optimal control variables are useful for several purposes. For example, to use the results in a statistically optimal sense one needs to quantify their errors to build their probability density distribution.

For Gaussian error distribution and in the linear approximation the posterior error covariance matrix \mathbf{P}^f of the control variables is the inverse Hessian matrix of the cost function J at its minimum, i.e. for the optimal set of control variables \mathbf{x}_{opt} (*Thacker, 1989*). The Hessian matrix is the second order derivative of the cost function.

$$(\mathbf{P}^f)^{-1} = \nabla_{\mathbf{x}}^2 J(\mathbf{x}_{opt}) = \nabla (\nabla_{\mathbf{x}} J(\mathbf{x}_{opt})) \quad (19)$$

The full error covariance matrix can be huge and, in general, cannot be computed with present available resources for sophisticated ocean or atmospheric models that have a large number of control variables ($O(10^6)$). However, products of this matrix with arbitrary vectors cost only about twice as much as an ADM integration. These products provide a module to extract some features of the Hessian matrix, e.g., the leading eigenvectors. Details about the number of operations for an ADM, TDM, and Hessian vector products compared to the number of operations for the cost function are given by *Griewank* (1993).

4. METHODS OF CONSTRUCTIONS

Applications described above obviously require a numerical code of the model, its adjoint, and its tangent linear. In the following we focus on adjoint code construction because it is much more complicated than the construction of the tangent linear model. A description of the construction of tangent linear models (forward mode of Computational Differentiation) is given by *Bischof et al.* (1992). The question is how practical coding of adjoint models can be done.

Suppose we want to simulate a dynamical system numerically. The development of a numerical simulation program is usually done in three steps. First, the analytical differential equations are formulated. Then a discretization scheme is chosen and the discrete equations are constructed. The last step is to implement an algorithm that solves the discrete equations in a programming language. The construction of the tangent linear and adjoint model code may start after any of these three steps.

4.1. Adjoint of analytical equations

The analytical model equations are transformed into the adjoint equations by applying the rules for analytical adjoint operators. These equations subsequently are discretized and solved using a numerical algorithm. However, since the product rule is not valid for discrete operators, one has to be careful in constructing the discrete adjoint operators. This method is mostly applied to box models having simple boundary conditions (*Schröter*, 1989).

4.2. Adjoint of discretized equations

Constructing the adjoint model from the discrete model equations is usually done by defining a Lagrange Function. The derivatives of the Euler-Lagrange equations with respect to the model variables yield the discrete adjoint equations. Applying this method, no ad-

joint operator has to be constructed explicitly. However, extensive and cumbersome coding is necessary. The boundary conditions are handled separately in most cases.

Thacker has introduced this concept into oceanography (*Thacker, 1987; Thacker and Long, 1988; Long and Thacker, 1989a,b*). and constructed the adjoint code of the GFDL ocean model this way (*Tziperman and Thacker, 1989; Tziperman et al., 1992a,b*).

4.3. Adjoint of model code

This article is concerned with the third method, where the adjoint code is developed directly from the numerical code of the model. A numerical model is an algorithm that can be viewed as a composition of differentiable functions \mathcal{F} , each representing a statement in the numerical code:

$$\mathbf{y} = \mathcal{F}(\mathbf{x}) := (\mathcal{F}_n \circ \mathcal{F}_{n-1} \circ \dots \circ \mathcal{F}_2 \circ \mathcal{F}_1)(\mathbf{x}) \quad (20)$$

with intermediate results:

$$\mathbf{z}^l := \mathcal{F}_l \circ \dots \circ \mathcal{F}_1(\mathbf{x}) \quad (21)$$

The composition is differentiated by application of the chain rule:

$$\mathcal{F}'(\mathbf{x}) = \mathcal{F}_n|_{\mathbf{z}^{n-1}} \cdot \mathcal{F}'_{n-1}|_{\mathbf{z}^{n-2}} \cdot \dots \cdot \mathcal{F}'_2|_{\mathbf{z}^1} \cdot \mathcal{F}'_1|_{\mathbf{x}} \quad (22)$$

The resulting multiple product of Jacobian matrices can be evaluated in any order, since matrix multiply is an associative operation¹. Operating in forward mode, the intermediate derivatives are computed in the same order as the model computes the composition. In contrast, the adjoint model operates in reverse mode, i.e. the intermediate derivatives are computed in reverse order. A detailed introduction to differentiation of algorithms is given by *Griewank (1989)*. This method is feasible even for highly sophisticated models with complicated boundary conditions.

In reverse mode, a distinct adjoint model code fragment corresponds to each model code statement. The adjoint code fragments are composed in reverse order compared to the model code. For each kind of statement simple rules can be formulated for constructing adjoint statements (*Talagrand, 1991; Thacker, 1991; Giering and Kaminski, 1998a*). This simplifies considerably the adjoint code construction and subsequent debugging. Two examples of adjoint code constructions are given in section A.

¹Note, matrix multiply does not commute.

4.4. *The TAMC: a source-to-source translator*

The Tangent linear and Adjoint Model Compiler (TAMC) is a source-to-source translator for Fortran programs (TAMC, R. Giering, unpublished manual, 1997). It generates Fortran routines for computation of the first-order derivatives out of Fortran routines computing a function. The derivatives are computed in the reverse mode (adjoint model) or in the forward mode (tangent-linear model). In both modes Jacobian-Matrix products can be computed. TAMC is an implementation of the rules described by *Giering and Kaminski* (1998a).

TAMC reads the program code and constructs an internal abstract representation. The code is checked for semantical correctness and several analyses are applied. Most importantly the data flow analysis detects all active variables: Given the independent and dependent variables and the top-level subroutine TAMC determines all variables which carry derivative information. Derivative code is only generated for those variables. An abstract representation to compute derivatives is generated and finally this is transformed to Fortran code.

The code generation can be influenced by compiler options and directives. In reverse mode the TAMC generates by default recalculations of required variables. Alternatively, these variables can be store and restored if specific compiler directives are provided in the code. Black-box routines for which the code is not available are handled by the TAMC if sufficient flow information about this routines is given in form of directives.

5. SENSITIVITIES

5.1. *Passive tracer in the ocean*

The MIT GCM solves the incompressible Navier-Stokes equations on a C-grid, with optional hydrostatic approximation. The model has been applied to a large range of scales of ocean dynamics ranging from studies of convective chimneys to global ocean circulation estimation (*Marshall et al.*, 1997b,a) and has been developed specifically for use on modern parallel computing platforms. For coarse resolution, global ocean circulation studies, mesoscale eddy transfer effects are achieved using schemes related to the parameterization of *Gent and McWilliams* (1990) but with spatially and temporally variable mixing coefficients (*Visbeck et al.*, 1997). A convective adjustment scheme is used to parameterize vertical mixing due to static instabilities.

For tracer simulations we use an “off-line” tracer model, based on the MIT ocean GCM (*Follows et al.*,

1996). Velocity, temperature, salinity and convective mixing parameters are stored periodically during a prognostic integration of the GCM, and used to drive the tracer model off-line. This represents a considerable economy in computational requirements and allows an efficient implementation of the tracer model using High Performance Fortran.

The off-line model for the tracer distribution, \bar{C} , takes the form:

$$\frac{\partial}{\partial t} \mathbf{C} + \nabla (\mathbf{u}^* \mathbf{C}) + \nabla (\mathbf{K} \nabla \mathbf{C}) + \mathbf{Q}_c = \mathbf{S} \quad (23)$$

where \mathbf{u}^* is the transformed Eulerian mean velocity (following *Gent and McWilliams (1990)*) that advects tracers and the tensor \mathbf{K} constrains sub grid scale mixing to be along isopycnal surfaces. Convective mixing in the off-line model, represented by \mathbf{Q}_c , uses the statistics of vertical convection events in the GCM to control appropriate vertical mixing of the off-line tracer. External sources and sinks (such as air-sea fluxes) are represented by \mathbf{S} . Applications of the off-line tracer model (in slightly different configurations) are illustrated in the publications *Follows et al. (1996)*; *Follows and Marshall (1996)*, and *Williams and Follows (1998)*.

For the sensitivity study provided here, the model domain is the North Atlantic and extends from 100W-10E and 5S-80N. The dependent variable is the concentration of a passive tracer at 75W,29N in the upper layer at a specific time. The adjoint of the off-line tracer model has been generated by TAMC and is integrated for one year. The sensitivity to concentrations a few months before have been monitored. One month earlier the sensitivity are still concentrated at one place but the diffusivity has already begun to broadened it (Fig. 3). If the adjoint model is integrated further (backward in time) the maximum of sensitivity starts to move upstream. (Fig. 4). This means the concentration in this area influences the final concentration at the specific point the most. But surprisingly 8 month earlier the largest sensitivity is found in the original area again. (Fig. 5). This can be understood by looking at the sensitivity in a deeper layer. (Fig. 6). Here the sensitivity is much larger than in the upper layer. Thus, in this case vertical mixing of sensitivities increases the sensitivity in an upper layer, where otherwise the sensitivity would be low, because it has been advected to other regions by the much faster velocities. In the first month the still large sensitivity has been mixed to lower layers and not moved very much because of the small velocities.

5.2. Passive tracer in the atmosphere

TM2 is a three-dimensional atmospheric transport model which solves the continuity equation for an arbitrary number of tracers on an Eulerian grid spanning the entire globe (Heimann, 1995). It is driven by stored meteorological fields derived from analyses of a weather forecast model. Tracer advection is calculated using the "slopes scheme" of Russel and Lerner (1981). Vertical transport due to convective clouds is parameterized by the cloud mass flux scheme of Tiedtke (1989). Source and sink processes are calculated for each tracer followed by the transports. The model has a regular grid horizontally ($8^\circ \times 10^\circ$) and sigma coordinates in the vertical (9 layers). The time-step of the model is 4 hours. The adjoint code has been generated by TAMC and is used for sensitivity studies (Kaminski et al., 1996) and data assimilation (Kaminski et al., 1998a,b).

In the present study the model is forced with meteorological fields of the year 1987, derived from analyses of the European Center for Medium Range Weather Forecast (ECMWF) updated every 12 hours. Only one passive tracer (CO_2) is modeled for the particular year. Here, the sensitivity of the December mean tracer concentration at the mountain Mauna Loa, Hawaii in December 1987 to monthly mean surface fluxes is of interest. This is a linear setup and the sensitivity are computed efficiently by means of the adjoint model because there is only one dependent variable and many independent.

The fields of sensitivity one to six months earlier are shown in Fig. 7. In December the concentration at Mauna Loa is most sensitive to surface sources around Hawaii. The main peak is East of the islands probably due to eastwards winds during this month. Another peak is in South Asia caused by the overall westerly winds in higher latitudes. The sensitivity to fluxes in November are distributed over the whole northern hemisphere. The southern hemisphere has still very little effect on the tracer concentration at Mauna Loa in December. The only exception is a tongue from Mauna Loa to the west coast of South America. Going further backward in time the two hemisphere are still different because there is little inter hemisphere exchange of air masses. Inside the hemispheres the sensitivities become more and more equally distributed by strong westerly winds.

5.3. North Atlantic meridional heat flux sensitivity

The adjoint of the MIT GCM described above has been generated by TAMC. A few technique code adaptations were necessary in order to apply TAMC. A sim-

plified checkpointing technique (*Griewank, 1992*) is applied to reduce the memory requirements for storing the model trajectory. This technique essentially divides the time interval the adjoint model is integrated into subintervals. For each of them the original model is integrated first to store intermediate results followed by the adjoint run which requires the intermediate results. The subintervals are processed in reverse order starting from previous stored *checkpoints* written by a previous model integration. In summary an additional model integration is required compared to a standard adjoint model integration and the memory requirements are reduced by about a square root of the time-steps. Without this technique, adjoint models of sophisticated ocean or atmospheric models could not be integrated on today's computers. Details of the adjoint model and its performance are described in (*J. Maroztke et al., submitted manuscript, 1999*). The adjoint model is used for data assimilation (*Stammer et al., 1997*) and sensitivity studies. Code to compute second order derivatives has been generated by TAMC (forward over reverse mode) and is used for error estimation of adjoint data assimilation results (*R. Giering, unpublished manuscript, 1998*).

The meridional North Atlantic heat flux and its variability is very important for climate in Europe. The sensitivity of the zonal integrated annual mean heat flux with respect to initial temperature and salinity of the year 1993 has been determined by the adjoint MIT GCM. The trajectory used for linearization has been computed by adjoint data assimilation (*Stammer et al., 1997*). The initial model state and the boundary forcing have been varied. The trajectory is an optimal fit to surface data (sea surface height from altimeter, wind stress, heat and fresh water fluxes), and subsurface data (analyzed temperature and salinity fields (*Levitus, 1989*)) which is also consistent with the model equations.

In Fig. 8 the sensitivity to salinity in 1160m depth is shown. The largest sensitivities can be found in the western boundary undercurrent. The water masses in this region are transported southwards across the section where we computed the heat flux. But salinity does not influence the flux of temperature directly, instead larger salinity means higher density and this influences the velocities. The higher the density inside the undercurrent the larger the southward velocity. Larger southward velocities of cold water masses increase the northward heat flux. This chain of reasons and causes is the explanation for the northward extend of sensitivities. Transport processes alone are too slow to bring water masses from the Labrador Sea down to 29N in

only one year.

6. CONCLUSIONS

An introduction into tangent linear (TLM) and adjoint models (ADM) was given. The theory is based on simple linear algebra, and the connection to tangent linear and adjoint operators has been made. Various applications of TLM's and ADM's have been explained.

TLM's and ADM's are the implementation of the forward and reverse mode of Computational Differentiation. Automatic differentiation is probably the most efficient way to construct these models especially for large and complex codes of sophisticated biogeochemical models. Several tools exist, they differ in the programming language they can handle and algorithmically. The TLM's and ADM's presented here have been generated by the Tangent linear and Adjoint Model Compiler (TAMC).

Sensitivities have been shown for three different biogeochemical models. The sensitivities can be explained by the processes in the corresponding system. Sensitivity fields provided by adjoint models can give a new insight into these processes. The interpretation of these fields is still in its infancy. But with upcoming studies in oceanography and meteorology the full potential of adjoint sensitivities will probably be examined. Tangent linear and adjoint models will become standard tools for the study of processes in all kinds of biogeochemical systems.

ACKNOWLEDGMENTS

The author thanks Mick Follows for his help running the MIT off-line ocean tracer model and Thomas Kaminski for providing sensitivity fields of the atmospheric tracer model TM2.

APPENDIX A: TAMC EXAMPLE

The construction of adjoint code by TAMC is demonstrated by the simple Fortran-90 subroutine (costfunc.f) shown below.

```
subroutine costfunc( n, x, fc )
integer n, i, niter
real x(n), fc, y(n)
y = x
do i = 1, niter
  y = sin( y )
end do
fc = sum( y*y )
end
```

The top-level routine is `costfunc`, it does not call any other subroutines. The dependent variable is `fc`, the value of a cost function and the independent variable is the vector of control variables `x`. The adjoint code (reverse mode) is generated by the command:

```
tamc -module costfunc -input x -output fc
      -reverse -pure -f90 costfunc.f
```

The option `pure` suppresses the computation of the cost function itself. Declarations in the adjoint code (`costfunc_ad.f`) have been removed and comments were added for demonstration purposes.

```
subroutine adcostfunc( n, x, adx, adfc )
--- declarations ---
ady(:) = 0           !reset local adjoints
y = x               !recompute last y
do i = 1, niter     !..
  y = sin(y)        !..
end do              !..
ady = ady+2*adfc*y !adjoint to
adfc = 0            !..fc = sum( y*y )
do i = niter, 1, -1
  y = x             !recomputation
  do i1 = 1, i-1    !..of actual y
    y = sin(y)      !..
  end do            !..
  ady = ady*cos(y) !adjoint loop kernel
end do
adx = adx+ady       !adjoint to
ady = 0              !..y = x
end
```

First of all all local adjoint variables are reset. Then the final value of `y` is recomputed for the following adjoint assignment. The adjoint loop operates in reverse order and every path begins with the recomputation of the required intermediate value of `y`. This recomputation is itself a loop and it is very expensive in terms of run-time.

In the second example (`costfunc2.f`) a few directives have been inserted to make the code more efficient by avoiding recomputations.

```

      subroutine costfunc2( n, x, fc )
      integer n, i, niter
      real x(n), fc, y(n)
CADJ INIT ftape = 'tp'
CADJ INIT ctape = common, niter
      y = x
      do i = 1, niter
CADJ STORE y = ctape
          y = sin( y )
      end do
CADJ STORE y = ftape
      fc = sum( y*y )
      end

```

The first two directives define tapes where values can be stored. The first tape (`ftape`) is realized as a direct access file on disc and the second (`ctape`) as static memory with a fixed number of records (`niter`). The storage itself is done at the places where the other two directives occur.

The new adjoint code (`costfunc2_ad.f`) is generated by the command:

```

tamc -module costfunc2 -input x -output fc
      -reverse -pure -f90 costfunc2.f

```

Again the code shown below has been edited.

```

subroutine adcostfunc2( n, adx, adfc )
--- declarations ---
open(60,ACCESS='DIRECT',RECL=8)
ady(:) = 0           !reset local adjoints
read(60,REC=1) y     !restore last y
ady = ady+2*adfc*y   !adjoint to
adfc = 0             !..fc = sum( y*y )
do i = niter, 1, -1
  y(:) = yh(:,i)    !restore actual y
  ady = ady*cos(y)  !adjoint loop kernel
end do
adx = adx+ady       !adjoint to
ady = 0.             !..y = x
close(60)            !close files
end

```

In contrast to the first adjoint code the required intermediate results are now restored from the tapes. The file which implements the first tape is opened and closed at the beginning and end of the subroutine. Records are read from this tape by direct access. Because Input/Output operations are usually slow access to records on this tape is slow. The second tape is a static array (`yh`), its last dimension extend is the number of records to be stored. Records are stored and restored by assignments, a very fast operation.

Compared to the first adjoint code this one needs much less run-time. Intermediate results are restored

from different tapes where in the first example they are recomputed. Thus, the penalty for the faster code is the higher memory requirements. In practice there is a trade-off between run-time and memory resources which depends on many parameters of the computer architecture the code is running on.

*

References

- Bischof, C. H., A. Carle, G. F. Corliss, A. Griewank, and P. Hovland, ADIFOR: Generating derivative code from FORTRAN programs, *Sci. Progr.*, *1*, 11–29, 1992.
- Courtier, P., J.-N. Thepaut, and A. Hollingsworth, A strategy for operational implementation of 4-dvar using an incremental approach, *Quart. J. of the Royal Met. Soc.*, *120*, 1367–1387, 1994.
- Errico, R., What is an Adjoint Model, *Bull. Amer. Meteor. Soc.*, *78*, 2577–2591, 1997.
- Errico, R. M., and T. Vukicevic, Sensitivity analysis using an adjoint of the PSU-NCAR mesoscale model, *Mon. Wea. Rev.*, *120*, 1644–1660, 1992.
- Farrel, B., and P. Ioannou, Generalized stability theory. part i: Autonomous operators., *J. of Atmos. Sci.*, *53*, 2025–2040, 1996a.
- Farrel, B., and P. Ioannou, Generalized stability theory. part ii: Nonautonomous operators., *J. of Atmos. Sci.*, *53*, 2041–2053, 1996b.
- Follows, M., and J. Marshall, Modelling bomb-produced radiocarbon in the north atlantic, *J. of Geophys. Res.*, *22*, 577–582, 1996.
- Follows, M., R. Williams, and J. Marshall, Subduction of carbon in the north atlantic subtropical gyre, *J. of Marine Res.*, *54*, 605–630, 1996.
- Gent, P., and J. C. McWilliams, Isopycnal mixing in ocean general circulation models, *J. of Phys. Oceanogr.*, *20*, 150–155, 1990.
- Giering, R., and T. Kaminski, Recipes for Adjoint Code Construction, *ACM Trans. On Math. Software*, *24*, 437–474, 1998a.
- Giering, R., and T. Kaminski, Using tamc to generate efficient adjoint code: Comparison of hand written and automatically generated code for evaluation of

- first and second derivatives., in *Automatic differentiation for adjoint code generation*, edited by C. Faure, pp. 31–37, INRIA, Rapport de recherche 3555, 1998b.
- Griewank, A., On automatic differentiation, in *Mathematical Programming: Recent Developments and Applications*, edited by M. Iri and K. Tanabe, pp. 83–108, Kluwer Academic Publishers, Dordrecht, 1989.
- Griewank, A., Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation, *Optimization Methods and Software*, 1, 35–54, 1992.
- Griewank, A., Some bounds on the complexity of gradients, Jacobians, and Hessians, in *Complexity in Nonlinear Optimization*, edited by P. M. Pardalos, pp. 128–161, World Scientific Publishers, 1993.
- Hall, M. C. G., D. G. Cacuci, and M. E. Schlesinger, Sensitivity analysis of a radiative-convective model by the adjoint method, *J. Atmos. Sci.*, 39, 2038–2050, 1982.
- Heimann, M., The global atmospheric tracer model T-M2, *Technical Report No. 10*, Max-Planck-Institut für Meteorologie, Hamburg, Germany, 1995.
- Kaminski, T., R. Giering, and M. Heimann, Sensitivity of the seasonal cycle of CO₂ at remote monitoring stations with respect to seasonal surface exchange fluxes determined with the adjoint of an atmospheric transport model, *Phys. and Chem. of the Earth*, 21, 457–462, 1996.
- Kaminski, T., R. Giering, and M. Heimann, A coarse grid three dimensional global inverse model of the atmospheric transport, 1, adjoint model and jacobian matrix, 1998a, *J. of Geophys. Res.*, in press.
- Kaminski, T., M. Heimann, and R. Giering, A coarse grid three dimensional global inverse model of the atmospheric transport, 2, inversion of the transport of co2 in the 1980s, 1998b, *J. of Geophys. Res.*, in press.
- Langland, R., R. Elsberry, and R. Errico, Evaluation of physical process in an idealized extratropical cyclone using adjoint sensitivity, *Quart. J. of the Royal Met. Soc.*, 121, 1349–1386, 1995.
- Levitus, S., Interpentadal Variability of Temperature and Salinity at Intermediate Depths of the North Atlantic Ocean, *J. Geophys. Res.*, 94, 6091–6131, 1989.

- Lewis, J., and J. Derber, The use of adjoint equations to solve a variational adjustment problem with advective constraints., *Tellus, A*, 309–322, 1985.
- Long, R., and W. Thacker, Data assimilation into a numerical equatorial ocean model. Part I : The model and assimilation algorithm., *Dyn. Atmos. Oceans*, 13, 379–412, 1989a.
- Long, R., and W. Thacker, Data assimilation into a numerical equatorial ocean model. Part II : Assimilation experiments., *Dyn. Atmos. Oceans*, 13, 413–440, 1989b.
- Marshall, J., A. Adcroft, C. Hill, L. Perelman, , and C. Heisey, A finite-volume, incompressible navier stokes model for studies of the ocean on parallel computers, *J. of Geophys. Res.*, C3, 5753–5766, 1997a.
- Marshall, J., C. Hill, L. Perelman, and A. Adcroft, Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling, *J. of Geophys. Res.*, C3, 5733–5752, 1997b.
- Navon, I., Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography, *Dynamics of Atmospheres and Oceans*, 1997, special Issue in honor of Richard Pfeffer.
- Oldenborgh, G., G. Burgers, S. Venzke, C. Eckert, and R. Giering, Tracking down the delayed ENSO oscillator with an adjoint OGCM, 1999, *Mon. Wea. Rev.*, in press.
- Rabier, F., P. Courtier, and O. Talagrand, An application of adjoint models to sensitivity analysis, *Beitr. Phys. Atmos.*, 65, 177–192, 1992.
- Russel, G. L., and J. A. Lerner, A new finite-differencing scheme for the tracer transport equation, *J. Appl. Met.*, pp. 1483–1498, 1981.
- Schröter, J., Driving of non-linear time dependent ocean models by observations of transient tracer - a problem of constrained optimization., in *Ocean Circulation Models: Combining Data and Dynamics*, edited by D. Anderson and J. Willebrand, pp. 257–285, Kluwer Academic Publishers, 1989.
- Stammer, D., C. Wunsch, R. Giering, Q. Zhang, J. Marotzke, J. Marshall, and C. Hill, The Global Ocean Circulation estimated from TOPEX/POSEIDON Altimetry and a General Circulation Model, *Tech. Rep. 49*, Center for

- Global Change Science, Massachusetts Institute of Technology, 1997.
- Talagrand, O., The use of adjoint equations in numerical modelling of the atmospheric circulation, in *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, edited by A. Griewank and G. F. Corliss, pp. 169–180, SIAM, Philadelphia, Penn., 1991.
- Thacker, W., Three lectures on fitting numerical models to observations., *Tech. rep.*, GKSS Forschungszentrum Geesthacht GmbH, Geesthacht, Federal Republic of Germany, 1987.
- Thacker, W., The role of the Hessian Matrix in fitting Models to measurements., *J. Geophys. Res.*, *94*, 6177–6196, 1989.
- Thacker, W. C., Automatic differentiation from an oceanographer’s perspective, in *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, edited by A. Griewank and G. F. Corliss, pp. 191–201, SIAM, Philadelphia, Penn., 1991.
- Thacker, W. C., and R. B. Long, Fitting dynamics to data, *J. Geophys. Res.*, *93*, 1227–1240, 1988.
- Tiedtke, M., A comprehensive mass flux scheme for cumulus parameterization in large-scale models, *Mon. Weath. Rev.*, *117*, 1779–1800, 1989.
- Tziperman, E., and W. Thacker, An optimal control/adjoint equation approach to studying the ocean general circulation, *J. Phys. Oceanogr.*, *19*, 1471–1485, 1989.
- Tziperman, E., W. Thacker, R. Long, and S.-M. Hwang, Ocean data analysis using a general circulation model, I, simulations, *J. Phys. Oceanogr.*, *22*, 1434–1457, 1992a.
- Tziperman, E., W. Thacker, R. Long, S.-M. Hwang, and S. Rintoul, Ocean data analysis using a general circulation model, II, North Atlantic model, *J. Phys. Oceanogr.*, *22*, 1458–1485, 1992b.
- Visbeck, M., J. Marshall, T. Haine, and M. Spall, Specification of eddy transfer coefficients in coarse-resolution ocean circulation, *J. of Phys. Oceanogr.*, *27*, 381–402, 1997.
- Vukicevic, T., Optimal initial perturbations for two cases of extratropical cyclogenesis, *Tellus Series A*, *50*, 143–166, 1998.

- Webster, S., and B. Hopkins, Adjoint and singular vectors in a barotropic model, in *Workshop on Adjoint Applications in Dynamic Meteorology*, 1994.
- Williams, R., and M. Follows, Estimating new production over the north atlantic due to the climatological ekman supply of nutrients, *Deep Sea Res.*, *45*, 461–489, 1998.
- Xu, Q., Generalized adjoint for physical processes with parameterized discontinuities: Part i: Basic issues and heuristic examples, *J. Atmos. Sci.*, *53*, 1123–1142, 1996a.
- Xu, Q., Generalized adjoint for physical processes with parameterized discontinuities: Part ii: Vector formulations and matching conditions, *J. Atmos. Sci.*, *53*, 1143–1155, 1996b.
- Zou, X., Tangent linear and adjoint of "on-off" processes and their feasibility for use in 4-dimensional variational assimilation, *Tellus*, *49A*, 3–31, 1997.
- Zou, X., I. Navon, A. Barcilon, J. Whittaker, and D. Cacuci, An adjoint sensitivity study of blocking in a two-layer isentropic model, *Mon. Wea. Rev.*, *121*, 2833–2857, 1993.

Figure 1. Tangent linear model : advection and diffusion of perturbations

Figure 1. Tangent linear model : advection and diffusion of perturbations

Figure 2. Adjoint model : advection and diffusion of influence

Figure 2. Adjoint model : advection and diffusion of influence

Figure 3. Sensitivity of top layer (25m) tracer concentration to the top layer concentration one month earlier.

Figure 3. Sensitivity of top layer (25m) tracer concentration to the top layer concentration one month earlier.

Figure 4. Sensitivity of top layer (25m) tracer concentration to the top layer concentration 5 month earlier.

Figure 4. Sensitivity of top layer (25m) tracer concentration to the top layer concentration 5 month earlier.

Figure 5. Sensitivity of top layer (25m) tracer concentration to the top layer concentration 8 month earlier.

Figure 5. Sensitivity of top layer (25m) tracer concentration to the top layer concentration 8 month earlier.

Figure 6. Sensitivity of top layer (25m) tracer concentration to the top layer concentration 8 month earlier.

Figure 6. Sensitivity of top layer (25m) tracer concentration to the top layer concentration 8 month earlier.

Figure 7. Sensitivity of December mean concentration at Mauna Loa to monthly fluxes.

Figure 7. Sensitivity of December mean concentration at Mauna Loa to monthly fluxes.

Figure 8. Sensitivity of the annual mean, zonal integrated, meridional heat flux at 29N in the North Atlantic to salinity in 1160m depth at the beginning of the year (from J. Maroztke et al., submitted manuscript, 1999).

Figure 8. Sensitivity of the annual mean, zonal integrated, meridional heat flux at 29N in the North Atlantic to salinity in 1160m depth at the beginning of the year (from J. Maroztke et al., submitted manuscript, 1999).

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING

TANGENT LINEAR AND ADJOINT MODELS

GIERING