Second Order Information in Data Assimilation

Francois-Xavier Le Dimet

Laboratoire de Mod*é*lisation et Calcul
Universit*é* Joseph Fourier
38041 Grenoble cedex 9, FRANCE

I. M. Navon*

Dept. of Mathematics and Computational Science
and Information Technology , Florida State University
Tallahassee, FL 32306, USA

Dacian N. Daescu

The University of Iowa
Dept.of Mathematics 14 MLH
Iowa City, IA 52242

December 29, 2000

*Corresponding author: School of Computational Science and Information Technology, Florida State University, Tallahassee, FL32306-4120

Abstract

In variational data assimilation (VDA) for meteorological and/or oceanic models, the assimilated fields are deduced by combining the model and the gradient of a cost functional measuring discrepancy between model solution and observation, via a first order optimality system.

However existence and uniqueness of the VDA problem along with convergence of the algorithms for its implementation depend on the convexity of the cost function.

Properties of local convexity can be deduced by studying the Hessian of the cost function in the vicinity of the optimum thus the necessity of second order information to ensure a unique solution to the VDA problem.

In this paper we present a comprehensive review of issues related to second order analysis of the problem of VDA along with many important issues closely connected to it.

In particular we study issues of existence, uniqueness and regularization through second order properties. We then focus on second order information related to statistical properties and on issues related to preconditioning and optimization methods and second order VDA analysis. Predictability and its relation to the structure of the Hessian of the cost functional is then discussed along with issues of sensitivity analysis in the presence of data being assimilated. Computational complexity issues are also addressed and discussed.

Automatic differentiation issues related to second order information are also discussed along with the computational complexity of deriving the second order adjoint .

Finally an application aimed at illustrating the use of automatic differentiation for deriving the second order adjoint as well as the Hessian/vector product applied to minimizing a cost functional of a meteorological problem using the truncated-Newton method is presented. Results verifying numerically the computational cost of deriving the second order adjoint as well as results related to the spectrum of the Hessian of the cost functional are displayed and discussed.

1 Introduction

Data assimilation can be described as the ensemble of techniques for retrieving geophysical fields from different sources such as observations, governing equations, statistics, buoys, ..., etc.

Being heterogeneous in nature, quality and density these data sources have to be put together to optimally retrieve (the meaning of "optimal" has to be precisely defined) the geophysical fields. Due to its inherent operational tasks, meteorology has played an important role in the development of data assimilation techniques. An ever increasing amount of data and models are considered as an ensemble from which the optimal information should be extracted.

Behind most of the classical methods used in meteorology such as: optimal interpolation, variational methods, statistical estimation etc., there is a variational principle, i.e. the retrieved fields are obtained through the minimization of some functional depending on the various sources of information.

The retrieved fields are obtained trough some optimality condition which can be an Euler or Euler-Lagrange condition if the regularity conditions are satisfied. Since these conditions are first order conditions, it follows that they involve the first order derivatives of the functional which is minimized. In this sense, data assimilation techniques are first order methods.

But first order methods are only necessary conditions for optimality but not sufficient ones. To obtain sufficient conditions we need to proceed one step further and to introduce second order information.

By the same token, from the mathematical point of view sensitivity studies with respect to some parameter can be obtained trough Gateaux derivatives with respect to this parameter. Therefore if we seek the sensitivity of fields which have already been defined trough some first order conditions we will have to proceed one order of derivation further and in this sense our sensitivity studies require second order information.

The purpose of this review paper is to show how to obtain and how to use in an efficient way second order information in data assimilation. In a first part we will show how the second order derivative can be computed, primarily in a very general framework, then illustrate it with some examples. Then we will show how this second order information can be linked to the issues of uniqueness of a solution to the problem of data assimilation. This will be shown to be not only a mathematical consideration but also rather a practical issue whereby information can be extracted by studying second order information.

In a second part of the paper we will proceed to show how to derive sensitivity analysis from models and data. The analysis of the impact of uncertainties in the model and in the data provides essential links between purely deterministic methods (such as variational data assimilation) and stochastic methods (Kalman filter type data). We will then proceed to demonstrate how the link between these methods can be clearly understood trough use of second order information.

Researchers in other disciplines have carried out pioneering work using second order information. Work in seismology using second order information and applying it to obtain accurate Hessian/vector products for truncated -Newton minimization was carried out by Santosa and Symes(1988,1989) and by Symes (1990,1991,1993).

Reuther(1996) and Arian (1999) illustrated the importance of second order adjoint analysis for optimal control and shape optimization for inviscid aerodynamics.

Second order information was tackled in automatic differentiation (AD) by Abate et al.(1997), Giering and Kaminski (1998a, 1998b), Gay (1996) Hovland (1995), Griewank and Corliss (1991), Griewank (1993) and Griewank (2000) to cite but a few.

Several AD packages such as TAMC of Giering allow calculation of the Hessian of the cost functional.

Early work on second order information in meteorology includes Thacker (1989) followed by work of Wang et al. (1992), Wang et al. (1993), Wang (1993). Wang et al.(1995) and Wang et al. (1998) considered use of second order information for optimization purposes namely to obtain truncated -Newton and Adjoint Newton algorithms using exact Hessian/vector products. Application of these ideas was presented in Wang et al. (1997).

Kalnay et al.(2000) introduced an elegant and novel pseudo-inverse approach and showed its connection to the adjoint Newton algorithm of Wang et al. (1997). (See Kalnay et al. (2000), Pu and Kalnay (1999), Park and Kalnay (1999), Pu et al. (1997)).

Ngodock(1996) applied second order information in his doctoral thesis in conjunction with sensitivity analysis in the presence of observations and applied it to the ocean circulation. Le Dimet et al. (1997)presented the basic theory for second order adjoint analysis related to sensitivity analysis.

The structure of the paper is as follows. Section 2 deals with the theory of the second order adjoint method, both for time independent and time dependent models.The methodology is briefly illustrated using the shallow water equations model.

Section 3 deals with the connection between sensitivity analysis and second order information. Section 4 briefly presents the Kalnay et al. (2000) quasi-inverse method and its connection with second order information. Issues related to second order Hessian information in optimization theory are addressed in Section 5. Both unconstrained and constrained minimization issues are discussed. Finally the use of accurate Hessian/vector products to optimize the Truncated Newton method are presented along with the adjoint Truncated-Newton method.

A method for approximating the Hessian of the cost function with respect to the control variables proposed by Courtier et al. (1994), based on rank p approximation of it and bearing similarity to approximation of the Hessian in quasi-Newton methods is presented in Section 5.9.

Section 6 is dedicated to methods of obtaining the second order adjoint via automatic differentiation technology. Issues of computational complexity of A.D. for the second order adjoint are presented in Section 7.

Use of the Hessian of the cost functional to estimate error covariance matrices is briefly discussed in Section 8.

The use of Hessian singular vectors used for development of a simplified Kalman filter is addressed briefly in Section 9.

Finally as a numerical illustration we present in Section 10 the application of the second order adjoint of limited area model of the shallow water equations to obtain an accurate Hessian/vector product compared to an approximate Hessian vector product obtained by finite differences. Automatic differentiation is implemented using the adjoint model compiler TAMC. The Hessian/vector information is used in a truncated-Newton minimization algorithm of the cost functional with respect to the initial conditions taken as the control variables and its impact versus the Hessian/vector product obtained via finite differences is assessed.

The numerical results obtained verify the theoretically derived computational cost of obtaining the second order adjoint via automatic differentiation.

The ARPACK package was then used in conjunction with the second order adjoint to gain information about the spectrum of the Hessian of the cost function.

Summary and conclusions are presented in Section 11.

2 Computing the second order information

In this chapter we will deal with deterministic models while the case of stochastic modeling will be discussed later in this manuscript.

In general we will assume that the model has the general form:

$$F(\mathbf{X}, \mathbf{U}) = 0 \tag{1}$$

where $\mathbf{X}$, the state variable describes the state of the environment, $\mathbf{X}$ belongs to $\mathcal{H}$, which is in general a Hilbert space. $\mathbf{U}$ is the input of the model, i.e. an initial condition which has to be provided to the model to obtain from Eq. (1) a unique solution $\mathbf{X}(\mathbf{U})$. We will assume that $\mathbf{U} \in \mathcal{U}$ which is also equipped with a Hilbert space structure.

The closure of the model is obtained through a variational principle which can be considered as the minimization of some functional:

$$G(\mathbf{X}, \mathbf{U}) \tag{2}$$

For instance, in the case of variational data assimilation, G may be viewed as representing the cost function measuring the discrepancy between the observation and the solution associated with the value $\mathbf{U}$ of the input parameter. Therefore the optimal input for the model will minimize G.

2.1 First order necessary conditions

If the optimal $\mathbf{U}$ minimizes G, then it is given by the Euler equation which may be written as:

$$\nabla G(\mathbf{U}) = 0 \tag{3}$$

where ∇G is the gradient of G with respect to control variables.

The gradient of G is obtained in the following way:

(i) we compute the Gateaux (directional) derivative of the model and of G in some direction u. It comes that

$$\frac{\partial F}{\partial \mathbf{X}}\hat{\mathbf{X}} + \frac{\partial F}{\partial \mathbf{U}}u = 0 \tag{4}$$

where $\hat{()}$ stands for the Gateaux derivative. For a generic function Z it is given by:

$$\hat{Z}(\mathbf{U}) = \lim_{\alpha \to 0} \frac{Z(\mathbf{U} + \alpha u) - Z(\mathbf{U})}{\alpha} \tag{5}$$

$\frac{\partial F}{\partial \mathbf{X}}$ (or $\frac{\partial F}{\partial \mathbf{U}}$) is the Jacobian of F with respect to $\mathbf{X}$ (or $\mathbf{U}$) and

$$\hat{G}(\mathbf{X}, \mathbf{U}) = \left[\frac{\partial G}{\partial \mathbf{X}}, \hat{\mathbf{X}}\right] + \left[\frac{\partial G}{\partial \mathbf{U}}, u\right] \tag{6}$$

The gradient is obtained by exhibiting the linear dependence of $\hat{G}$ with respect to u. This is done by

(ii) we introduce the adjoint variable P(to be defined later according to convenience).

The inner product of P is taken, leading to

$$\left(\frac{\partial F}{\partial \mathbf{X}} \cdot \hat{\mathbf{X}}, P\right) + \left(\frac{\partial F}{\partial \mathbf{U}} \cdot u, P\right) = 0 \tag{7}$$

$$\left(\left(\frac{\partial F}{\partial \mathbf{X}}\right)^T \cdot P, \hat{\mathbf{X}}\right) + \left(\left(\frac{\partial F}{\partial \mathbf{U}}\right)^T \cdot P, u\right) = 0 \tag{8}$$

Therefore using (6), if P is defined as the solution of the adjoint model

$$\left(\frac{\partial F}{\partial \mathbf{X}}\right)^T \cdot P = \frac{\partial G}{\partial \mathbf{X}} \tag{9}$$

then we obtain

$$\nabla G(\mathbf{U}) = -\left(\frac{\partial F}{\partial \mathbf{U}}\right)^T \cdot P + \frac{\partial G}{\partial \mathbf{U}} \tag{10}$$

Therefore the gradient is computed by solving Eq. (9) to obtain P, then by applying Eq. (10).

2.2 Second order adjoint

To obtain second order information we seek for the product of the Hessian $H(\mathbf{U})$ of G with some vector u. As before we apply a perturbation u to Eqs. (1), (9), and from Eq. (9) and (10) we obtain

$$\left[\frac{\partial^2 F}{\partial \mathbf{X}^2} \cdot \hat{\mathbf{X}} + \frac{\partial^2 F}{\partial \mathbf{X} \partial \mathbf{U}} \cdot u\right]^T \cdot P + \left[\frac{\partial F}{\partial \mathbf{X}}\right]^T \hat{P} = \frac{\partial^2 G}{\partial \mathbf{X}^2} \cdot \hat{\mathbf{X}} + \frac{\partial^2 G}{\partial \mathbf{X} \partial \mathbf{U}} \cdot u \tag{11}$$

and

$$\begin{aligned} \nabla \widehat{G(\mathbf{U})} &= H(\mathbf{U}) \cdot u = \\ &= -\left[\frac{\partial^2 F}{\partial \mathbf{U}^2} \cdot u + \frac{\partial^2 F}{\partial \mathbf{U} \partial \mathbf{X}}\right]^T \cdot P - \left[\frac{\partial F}{\partial \mathbf{U}}\right]^T \cdot \hat{P} + \frac{\partial^2 G}{\partial \mathbf{U}^2} u + \frac{\partial^2 G}{\partial \mathbf{U} \partial \mathbf{X}} \hat{\mathbf{X}} \end{aligned} \tag{12}$$

We introduce here Q and R, two additional variables. To eliminate $\hat{\mathbf{X}}$ and $\hat{P}$, we will take the inner product of Eq. (4) and (11) with Q and R respectively, then add the results. We then obtain

$$\left(\hat{\mathbf{X}}, \left(\frac{\partial F}{\partial \mathbf{X}}\right)^T \cdot Q\right) + \left(u, \left(\frac{\partial F}{\partial \mathbf{U}}\right)^T \cdot Q\right) + \left(P, \frac{\partial^2 F}{\partial \mathbf{X}^2} \cdot \hat{\mathbf{X}} \cdot R\right) +$$

$$
\begin{aligned}
&+ \left(P, \frac{\partial^2 F}{\partial \mathbf{X} \partial \mathbf{U}} \cdot u \cdot R\right) + \left(\hat{P}, \left(\frac{\partial F}{\partial \mathbf{X}}\right) \cdot R\right) = \\
&= \left(\hat{\mathbf{X}}, \left(\frac{\partial^2 G}{\partial \mathbf{X}^2}\right)^T \cdot R\right) + \left(u, \left(\frac{\partial^2 G}{\partial \mathbf{X} \partial \mathbf{U}}\right)^T \cdot R\right)
\end{aligned} \tag{13}
$$

Let us take the inner product of Eq. (12) with u, then it comes

$$
\begin{aligned}
(H(\mathbf{U}) \cdot u, u) &= \left(-\left[\frac{\partial^2 F}{\partial \mathbf{U}^2} \cdot u + \frac{\partial^2 F}{\partial \mathbf{U} \partial \mathbf{X}}\right]^T \cdot P, u\right) + \\
&+ \left(\hat{P}, \left[-\frac{\partial F}{\partial \mathbf{U}}\right] \cdot u\right) + \left(\frac{\partial^2 G}{\partial \mathbf{U}^2} \cdot u, u\right) + \left(\hat{\mathbf{X}}, \left(\frac{\partial^2 G}{\partial \mathbf{U} \partial \mathbf{X}}\right)^T \cdot u\right)
\end{aligned} \tag{14}
$$

but

$$
\left(P, \frac{\partial^2 F}{\partial \mathbf{X}^2} \hat{\mathbf{X}} \cdot R\right) = \left(\hat{\mathbf{X}}, \frac{\partial^2 F}{\partial \mathbf{X}^2} P \cdot R\right) \tag{15}
$$

and

$$
\left(P, \frac{\partial^2 F}{\partial \mathbf{X} \partial \mathbf{U}} u \cdot R\right) = \left(u, \frac{\partial^2 F}{\partial \mathbf{X} \partial \mathbf{U}} P \cdot R\right) \tag{16}
$$

From (13) we get

$$
\begin{aligned}
&\left(\hat{\mathbf{X}}, \left(\frac{\partial F}{\partial \mathbf{X}}\right)^T Q + \frac{\partial^2 F}{\partial \mathbf{X}^2} P \cdot R - \left(\frac{\partial^2 G}{\partial \mathbf{X}^2}\right)^T \cdot R\right) + \left(\hat{P}, \frac{\partial F}{\partial \mathbf{X}} \cdot R\right) \\
&\quad = \left(u, -\left(\frac{\partial F}{\partial \mathbf{U}}\right)^T \cdot Q - \frac{\partial^2 F}{\partial \mathbf{X} \partial \mathbf{U}} P \cdot R + \left(\frac{\partial^2 G}{\partial \mathbf{X} \partial \mathbf{U}}\right)^T R\right)
\end{aligned} \tag{17}
$$

Therefore if Q and R are defined as being the solution of

$$
\left(\frac{\partial F}{\partial \mathbf{X}}\right)^T Q + \frac{\partial^2 F}{\partial \mathbf{X}^2} P \cdot R - \left(\frac{\partial^2 G}{\partial \mathbf{X}^2}\right)^T \cdot R = \left(\frac{\partial^2 G}{\partial \mathbf{X} \partial \mathbf{U}}\right)^T u \tag{18}
$$

$$
\left(\frac{\partial F}{\partial \mathbf{X}}\right) R = -\left(\frac{\partial F}{\partial \mathbf{U}}\right) u \tag{19}
$$

then we obtain:

$$
\begin{aligned}
(H(\mathbf{U}) \cdot u) &= -\left(\frac{\partial^2 F}{\partial \mathbf{U}^2} \cdot u + \frac{\partial^2 F}{\partial \mathbf{U} \partial \mathbf{X}}\right) \cdot P + \frac{\partial^2 G}{\partial \mathbf{U}^2} \cdot u \\
&- \left(\frac{\partial F}{\partial \mathbf{U}}\right)^T Q - \left(\frac{\partial^2 F}{\partial \mathbf{X} \partial \mathbf{U}} \cdot P\right) R + \frac{\partial^2 G}{\partial \mathbf{X} \partial \mathbf{U}} R
\end{aligned} \tag{20}
$$

The system (18)- (19) will be called the second order adjoint. Therefore we can obtain the product of the Hessian by a vector u by

(i) solving the system (18)- (19).

(ii) applying formula (20).

2.3 Remarks

a) The system (18)- (19) which has to be solved to obtain the Hessian $\times$ vector product can be derived from the Gateaux derivative (4) which is the same as (19). In the literature, this system is often called the linear tangent model, this denomination being rather inappropriate because it implies the issue of linearization and the subsequent notion of range of validity which is not relevant in the case of a derivative.

b) In the case of an N-finite dimensional space then the Hessian can be fully computed after N integrations of vector e_i of the canonical base.

Equation (18) differs from the adjoint model by the forcing terms which will depend on u and R.

c) The system (18), (19)- (20) will yield the exact value of the Hessian/vector product. An approximation could be obtained by standard finite differences, i.e.,

$$H(\mathbf{U}) \cdot u \simeq \frac{1}{\alpha}\left[\nabla G(\mathbf{U} + \alpha u) - \nabla G(\mathbf{U})\right] \tag{21}$$

where α is the finite-difference interval which has to be judicially chosen.

However several integrations of the model and of its adjoint model will be necessary in this case to determine the range of validity of the finite-difference approximation (Wang 1995 and references therein).

2.4 Time dependent model

In the case of variational data assimilation the model F is a differential system on the time interval $[0, T]$. The evolution of $X \in \mathcal{H} \subset [C(0,T)]^n$ between 0 and T is governed by the differential system,

$$\frac{d\mathbf{X}}{dt} = F(\mathbf{X}) + B\mathbf{V} \tag{22}$$

The input variable is often the initial condition,

$$\mathbf{X}(0) = \mathbf{U} \in \mathcal{R}^n \tag{23}$$

In this system F is a nonlinear operator which describes the dynamics of the model, $\mathbf{V} \in \mathcal{V} \subset [C(0,T)]^m$ is a term used to represent the uncertainties of the model, $\mathbf{U}$ is the initial condition, and the criteria G is the discrepancy between the solution of (22)-(23) and observations

$$J(\mathbf{U},\mathbf{V}) = \frac{1}{2}\int_0^T \|C\mathbf{X} - \mathbf{X}_{obs}\|^2 \, dt \tag{24}$$

where C is the observation matrix, i.e., a linear operator mapping $\mathbf{X}$ into O_{obs}. The problem consists in determining $\mathbf{U}$ and $\mathbf{V}$ that minimize J.

It is well know that in this case the Gateaux derivatives and adjoint model may be written as

$$\frac{dP}{dt} + \left[\frac{\partial F}{\partial \mathbf{X}}\right]^T \cdot P = C^T(C\mathbf{X} - \mathbf{X}_{obs}) \tag{25}$$

$$P(T) = 0 \tag{26}$$

and the components of the gradient ∇J with respect to $\mathbf{U}$ and $\mathbf{V}$ are

$$\nabla J_U = -P(0) \tag{27}$$

$$\nabla J_V = -B^T P \tag{28}$$

Let h be a perturbation on the control variables $\mathbf{U}$ and $\mathbf{V}$

$$h = \begin{pmatrix} h_U \\ h_V \end{pmatrix} \tag{29}$$

The Gateaux derivatives $\hat{\mathbf{X}}$, $\hat{P}$ of $\mathbf{X}$ and P in the direction of h, are obtained as the solution of the coupled system

$$\frac{d\hat{\mathbf{X}}}{dt} = \frac{\partial F}{\partial \mathbf{X}}\hat{\mathbf{X}} + Bh_V \tag{30}$$

$$\widehat{\mathbf{X}(0)} = h_U \tag{31}$$

$$\frac{d\hat{P}}{dt} + \left[\frac{\partial^2 F}{\partial \mathbf{X}^2} \cdot \hat{\mathbf{X}}\right]^T \cdot P + \left[\frac{\partial F}{\partial \mathbf{X}}\right]^T \cdot \hat{P} = C^T \cdot C\hat{\mathbf{X}} \tag{32}$$

$$\hat{P}(T) = 0 \tag{33}$$

$$\widehat{\nabla J_U} = -\hat{P}(0) \tag{34}$$

$$\widehat{\nabla J_V} = -B^T \hat{P} \tag{35}$$

Taking the inner product of (30) with Q and of (32) with R, integrating from 0 to T, then adding the resulting equations, it comes:

$$\int_0^T \left[\left(\frac{d\hat{\mathbf{X}}}{dt}, Q \right) - \left(\frac{\partial F}{\partial \mathbf{X}} \cdot \hat{\mathbf{X}}, Q \right) - (Bh_V, Q) + \left(\frac{d\hat{P}}{dt}, R \right) + \left(\left[\frac{\partial^2 F}{\partial \mathbf{X}^2} \cdot \hat{\mathbf{X}} \right]^T \cdot P, R \right) + \right.$$
$$\left. + \left(\left[\frac{\partial F}{\partial \mathbf{X}} \right]^T \cdot \hat{P}, R \right) - \left(C^T \cdot C\hat{\mathbf{X}}, R \right) \right] dt = 0 \tag{36}$$

After integration by parts and some additional transformations we obtain

$$\int_0^T \left(\hat{\mathbf{X}}, -\frac{dQ}{dt} - \left[\frac{\partial F}{\partial \mathbf{X}} \right]^T \cdot Q + \left[\frac{\partial^2 F}{\partial \mathbf{X}^2} P \right]^T \cdot R - C^T C R \right) dt$$
$$+ \int_0^T \left(\hat{P}, -\frac{dR}{dt} + \left(\frac{\partial F}{\partial \mathbf{X}} \right) R \right) dt - \int_0^T (h_V, B^T Q) dt$$
$$+ \left(\hat{\mathbf{X}}(T), Q(T) \right) - \left(\hat{\mathbf{X}}(0), Q(0) \right) + \left(\hat{P}(T), R(T) \right) - \left(\hat{P}(0), R(0) \right) = 0 \tag{37}$$

Let H be the Hessian matrix of the cost J. We have

$$H = \begin{pmatrix} H_{UU} & H_{UV} \\ H_{VU} & H_{VV} \end{pmatrix} \tag{38}$$

Therefore if we define the second order adjoint as being the solution of

$$\frac{dQ}{dt} + \left[\frac{\partial F}{\partial \mathbf{X}} \right]^T Q = \left[\frac{\partial^2 F}{\partial \mathbf{X}^2} \cdot P \right]^T \cdot R - C^T C R \tag{39}$$
$$\frac{dR}{dt} = \left[\frac{\partial F}{\partial \mathbf{X}} \right] R \tag{40}$$

it means that if we take

$$Q(T) = 0 \tag{41}$$
$$R(0) = h_U \tag{42}$$

then it remains

$$-(h_U, Q(0)) = \left(\hat{P}(0), R(0) \right) \tag{43}$$

$$\hat{P}(0) = -Q(0) \tag{44}$$

The product of the Hessian by a vector is obtained exactly by a direct integration of (40)- (42) followed by a backward integration in time of (39)- (41).

One can obtain H by an integration of the differential system

$$\begin{aligned} \frac{dQ}{dt} + \left[\frac{\partial F}{\partial X}\right]^T \cdot Q &= \left[\frac{\partial^2 F}{\partial X^2} \cdot P\right]^T \cdot R - C^T C R & (45) \\ \frac{dR}{dt} &= \left[\frac{\partial F}{\partial \mathbf{X}}\right] R & (46) \end{aligned}$$

with the conditions

$$Q(T) = 0 \tag{47}$$

$$R(0) = e_i \tag{48}$$

where e_i is the n-vector of the canonical base of R^n obtaining

$$H_{UU} e_i = Q(0) \tag{49}$$

$$H_{UV} e_i = B^T Q \tag{50}$$

One then integrates m times the differential system

$$\frac{dQ}{dt} + \left[\frac{\partial F}{\partial X}\right]^T \cdot Q = \left[\frac{\partial^2 F}{\partial X^2} \cdot P\right]^T \cdot R - C^T C R \tag{51}$$

$$\frac{dR}{dt} - \left[\frac{\partial F}{\partial X}\right] R = f_j \tag{52}$$

with initial and terminal conditions

$$Q(T) = 0 \tag{53}$$

$$R(0) = 0 \tag{54}$$

where f_j are the m canonical base vectors of R^m obtaining

$$H_{VV} \cdot f_j = B^T Q, \tag{55}$$

The system defined by these equations is the second order adjoint model. The Hessian matrix is obtained via $n + m$ integrations of the second order adjoint. The second order adjoint is easily obtained from the first order adjoint - differing from it by some forcing terms, in particular the second order term. The second equation is that of the linearized model (the tangent linear model).

2.5 Example: The shallow-water equations

The shallow-water equations (SWE) represent the flow of an incompressible fluid whose depth is small with respect to the horizontal dimension.

The SWE can be written in a Cartesian system

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} - fv + \frac{\partial \phi}{\partial x} = 0 \tag{56}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + fu + \frac{\partial \phi}{\partial y} = 0 \tag{57}$$

$$\frac{\partial \phi}{\partial t} + \frac{\partial u\phi}{\partial x} + \frac{\partial v\phi}{\partial y} = 0 \tag{58}$$

In this system of equations $X = (u, v, \phi)^T$ is the state variable, u and v are the components of the horizontal velocity,ϕ is the geopotential and f the Coriolis parameter. We aim to present this example in order to provide a didactic setup, thus we will make the strongest simplifications.

a) We neglect the model error which following the previous notations implies $B \equiv 0$. We only control the initial conditions.

b) We impose periodic boundary conditions.

c) The observations are assumed continuous in both space and time, which is tantamount to assume $C \equiv I$, where I is the identity operator. Let $U_0 = (u_0, v_0, \phi_0)^T$, i.e., the initial condition, then the cost function assume the form

$$J(U_0) = \frac{1}{2}\int_0^T [(u - u_{obs})^2 + (v - v_{obs})^2 + (\phi - \phi_{obs})^2]dt. \tag{59}$$

We derive directly the tangent linear model (TLM). The barred variables $\bar{X} = (\bar{u}, \bar{v}, \bar{\phi})^T$ are the directional derivative in the direction of the perturbation $h = (h_u, h_v, h_\phi)^T$ applied to the initial condition and we obtain

$$\frac{\partial \bar{u}}{\partial t} + u\frac{\partial \bar{u}}{\partial x}\bar{u}\frac{\partial u}{\partial x} + v\frac{\partial \bar{u}}{\partial y} + \bar{v}\frac{\partial u}{\partial y} - f\bar{v} + \frac{\partial \bar{\phi}}{\partial x} = 0 \tag{60}$$

$$\frac{\partial \bar{v}}{\partial t} + u\frac{\partial \bar{v}}{\partial x} + \bar{u}\frac{\partial v}{\partial x} + v\frac{\partial \bar{v}}{\partial y} + \bar{v}\frac{\partial v}{\partial y} + f\bar{u} + \frac{\partial \bar{\phi}}{\partial y} = 0 \tag{61}$$

$$\frac{\partial \bar{\phi}}{\partial t} + \frac{\partial \bar{u}\phi}{\partial x} + \frac{\partial u\bar{\phi}}{\partial x} + \frac{\partial \bar{v}\phi}{\partial y} + \frac{\partial v\bar{\phi}}{\partial y} = 0 \tag{62}$$

By transposing the TLM we obtain the adjoint model. Let $P = (\tilde{u}, \tilde{v}, \tilde{\phi})^T$ be the adjoint variable, then the adjoint model satisfies

$$\frac{\partial \tilde{u}}{\partial t} + u\frac{\partial \tilde{u}}{\partial x} + v\frac{\partial \tilde{v}}{\partial y} + \tilde{u}\frac{\partial v}{\partial y} - \tilde{v}\frac{\partial v}{\partial y} - f\tilde{v} + \phi\frac{\partial \tilde{\phi}}{\partial x} = u_{obs} - u \tag{63}$$

$$\frac{\partial \tilde{v}}{\partial t} + \tilde{u}\frac{\partial u}{\partial y} - u\frac{\partial \tilde{v}}{\partial x} + \tilde{v}\frac{\partial u}{\partial x} + v\frac{\partial \tilde{v}}{\partial y} + f\tilde{u} + \phi\frac{\partial \tilde{\phi}}{\partial y} = v_{obs} - v \tag{64}$$

$$\frac{\partial \tilde{\phi}}{\partial t} + \frac{\partial \tilde{u}}{\partial x} + \frac{\partial \tilde{v}}{\partial y} + u\frac{\partial \tilde{\phi}}{\partial x} + v\frac{\partial \tilde{\phi}}{\partial y} = \phi_{obs} - \phi. \tag{65}$$

To obtain the second order model we linearize the couple direct model and adjoint model, we then transpose and obtain the second order adjoint variable $Q = (\hat{u}, \hat{v}, \hat{\phi})^T$ and the variable $R = (\bar{u}, \bar{v}, \bar{\phi})^T$ defined by the TLM.

$$\begin{aligned}\frac{\partial \hat{u}}{\partial t} + u\frac{\partial \hat{u}}{\partial x} + v\frac{\partial \hat{v}}{\partial y} + \hat{u}\frac{\partial v}{\partial y} - \hat{v}\frac{\partial v}{\partial y} - f\hat{v} + \phi\frac{\partial \hat{\phi}}{\partial x}&\\ = \tilde{v}\frac{\partial \bar{v}}{\partial x} - \bar{u}\frac{\partial \tilde{u}}{\partial x} - \bar{v}\frac{\partial \bar{u}}{\partial y} + \tilde{u}\frac{\partial \bar{v}}{\partial y} - \bar{\phi}\frac{\partial \tilde{\phi}}{\partial x} - \bar{u}&\end{aligned} \tag{66}$$

$$\begin{aligned}\frac{\partial \hat{v}}{\partial t} + \hat{u}\frac{\partial u}{\partial y} - u\frac{\partial \hat{v}}{\partial x} + \hat{v}\frac{\partial u}{\partial x} + v\frac{\partial \hat{v}}{\partial y} + f\hat{u} + \phi\frac{\partial \hat{\phi}}{\partial y}&\\ = \tilde{u}\frac{\partial \bar{u}}{\partial x} - \bar{u}\frac{\partial \tilde{v}}{\partial x} - \tilde{v}\frac{\partial \bar{u}}{\partial y} + \bar{u}\frac{\partial \tilde{v}}{\partial y} - \bar{\phi}\frac{\partial \tilde{\phi}}{\partial y} - \bar{v}&\end{aligned} \tag{67}$$

$$\frac{\partial \hat{\phi}}{\partial t} + \frac{\partial \hat{u}}{\partial x} + \frac{\partial \hat{v}}{\partial y} + u\frac{\partial \hat{\phi}}{\partial x} + v\frac{\partial \hat{\phi}}{\partial y} = -\bar{u}\frac{\partial \tilde{\phi}}{\partial x} - \bar{v}\frac{\partial \tilde{\phi}}{\partial x} - \bar{\phi} \tag{68}$$

We see that formally the first and second order adjoint models differ only by second order terms, which contain the adjoint variables.

The calculation of second order derivatives requires the storage of the model trajectory, the tangent linear model, and the adjoint model.

3 Sensitivity analysis and second order information

In general we can consider that a model has three kinds of variables: *state variables* Z in a space $\mathcal{L}$ which are the values of the variables describing the medium (wind, temperature, pressure, humidity, ...); *input variables* I in a space $\mathcal{I}$ which have to be provided to the model: initial and/or boundary conditions. Most of the time these input variables are not directly plugged (inserted) into the model, but derived from observations through a method of data assimilation; *parameters* K: most of the models employ empirical parameters (e.g. diffusivity) which have to be tuned through some validation process prior to using the model. Therefore from a mathematical point of view, the model is written as:

$$\mathcal{F}(Z, I, K) = 0 \tag{69}$$

where $\mathcal{F}$ is some PDE operator or its discrete form. We assume that I and K are prescribed (or I and K being prescribed), in which case defined by using an additional functional $G : \mathcal{L} \to R$ called a response function. By definition, the sensitivity of the model with respect to the input I (or the parameter K) is the gradient of the response function with respect to I (respectively K). The difficulty comes from the fact that I is implicitly used in G through Z. To compute the sensitivity we derive the model in a direction h on I and a direction j on K

$$\frac{\partial \mathcal{F}}{\partial Z}\hat{Z} + \frac{\partial \mathcal{F}}{\partial I}h + \frac{\partial \mathcal{F}}{\partial K}j = 0 \tag{70}$$

The directional derivative of G is

$$\hat{G}(Z, h, j) = \frac{\partial G}{\partial Z}\hat{Z}$$

Computing the gradient of G with respect to I and K requires to exhibit the linear dependence of $\hat{G}$ with respect to h and j. Let us now introduce D, an adjoint variable to be defined later for convenience. We take the inner product of (70) with D obtaining

$$\left(\hat{Z}, \left[\frac{\partial \mathcal{F}}{\partial Z}\right]^T \cdot D\right) + \left(h, \left[\frac{\partial \mathcal{F}}{\partial I}\right]^T \cdot D\right) + \left(j, \left[\frac{\partial \mathcal{F}}{\partial K}\right]^T \cdot D\right) = 0$$

Clearly, if D is defined as the solution of

$$\left(\frac{\partial \mathcal{F}}{\partial Z}\right)^T \cdot D = \frac{\partial G}{\partial Z} \tag{71}$$

then the identification yields

$$\hat{G}(Z,h,j) = -\left(h, \left[\frac{\partial \mathcal{F}}{\partial I}\right]^T \cdot D\right) - \left(j, \left[\frac{\partial \mathcal{F}}{\partial K}\right]^T \cdot D\right) \tag{72}$$

from which we get

$$\nabla_I G = -\left[\frac{\partial \mathcal{F}}{\partial I}\right]^T \cdot D \tag{73}$$

$$\nabla_K G = -\left[\frac{\partial \mathcal{F}}{\partial K}\right]^T \cdot D \tag{74}$$

Therefore the sensitivity is estimated from (73) and (74) with D the solution of the adjoint model (71).

The actual input of a model consists of observations which are transformed after some more or less complicated process into the mathematical input. In the variational data assimilation the discrepancy between the observational data X_{obs} and the solution of the model

$$F(X,I,K) = 0$$

is measured by a cost function J which may be defined as

$$J(I) = \frac{1}{2}\|C \cdot X(I,K) - X_{obs}\|^2$$

In the variational data assimilation method the observations appear then only in the right-hand side of the adjoint model as a forcing term

$$\left[\frac{\partial F}{\partial X}\right]^T \cdot P = \frac{\partial J}{\partial X} \tag{75}$$

Therefore in order to derive the sensitivity with respect to the observations it will be necessary to generalize the concept of model to include the adjoint model (75). The generalized variable will be

$$Z = \begin{pmatrix} X \\ P \end{pmatrix} \tag{76}$$

and the generalized model

$$\mathcal{F}(Z,I,K) = \begin{pmatrix} F(X,I,K) \\ \left(\frac{\partial F}{\partial X}\right)^T \cdot P - \frac{\partial J}{\partial X} \end{pmatrix} = 0 \tag{77}$$

The sensitivity will have to be derived on this system, therefore the adjoint of the optimality system will have to be derived. Clearly, the sensitivity analysis depends now on second order information. For an in-depth discussion see LeDimet et. al (1997).

4 Kalnay et al.(2000) quasi inverse method and second order information

The inverse 3-D Var proposed by Kalnay et al.(2000) it is introduced by considering a cost functional

$$J = \frac{1}{2}(L\delta x)^T B^{-1}(L\delta x) + \frac{1}{2}[HL\delta x - \delta y]^T R^{-1}[HL\delta x - \delta y] \tag{78}$$

where δx is the difference between the analysis and the background at the beginning of the assimilation window, L and L^T are the TLM and its adjoint, and H is the tangent linear version of the forward observation operator $\mathbf{H}$. B is the forecast error covariance and R is the observational error covariance.

Taking the gradient of J with respect to the initial change $\delta x = x^a - x^b$, where x^a and x^b are the analysis and first guess respectively, we obtain

$$\nabla J = L^T(B^{-1}L\delta x + H^T R^{-1}[HL\delta x - \delta y]) \tag{79}$$

In an adjoint 4-D Var an iterative minimization algorithm is employed to obtain the optimal perturbation:

$$\delta x^i = \alpha_i \nabla J^{i-1} \tag{80}$$

where α_i is the step in the minimization algorithm.

The inverse 3-D Var approach of Kalnay seeks to obtain directly the "perfect solution", i.e. the special δx that makes $\nabla J = 0$, provided δx is small.

Eliminating in (79) the adjoint operator one gets

$$L\delta x = (B^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} \delta y \tag{81}$$

Since we have the quasi-inverse model obtained by integrating TLM backwards, i.e. a good approximation of L^{-1}, we obtain:

$$\delta x = L^{-1}(B^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} \delta y \tag{82}$$

As shown by Kalnay et al. (2000) this is equivalent to the Adjoint Newton Algorithm by Wang et al. (1997) except that it does not require a line minimization.

Wang et al. (1998) proposed an adjoint Newton algorithm which also required the backwards integration of the tangent linear model and proposed a reformulation of the adjoint Newton when the TLM is not invertible.They did not explore this idea in depth. To show the link of inverse 3-D Var to second order information we follow Kalnay et al. (2000) to show that inverse 3-D Var is equivalent to using a perfect Newton iterative method to solve the minimization problem at a given time level.

If we look for the minimum of the cost functional at $x + \delta x$ given that our present estimate of the solution is x then expanding $\nabla J(x+\delta x)$ in a Taylor series to second term yields

$$\nabla J(x + \delta x) = \nabla J(x) + \nabla^2 J(x) \delta x = 0 \tag{83}$$

where $\nabla^2 J(x)$ is the Hessian matrix.

The Newton iteration is

$$\delta x = -[\nabla^2 J(x)]^{-1} \nabla J(x) \tag{84}$$

For the cost function (78) the Hessian is given by

$$\nabla^2 J(x) = L^T [B^{-1} + H^T R^{-1} H] L \tag{85}$$

A first iteration with the Newton minimization algorithm yields

$$\delta x_1 = [L^T (B^{-1} + H^T R^{-1} H) L]^{-1} L^T H^T R^{-1} \delta y \tag{86}$$

which is identical with the inverse 3-D Var solution.

Since cost functions used in 4-D Var are close to quadratic functions one may view 3-D Var as a perfect preconditioner of a simplified 4-D Var problem.

In general availability of second order information allows powerful minimization algorithms to perform (Wang et al. 1995, Wang et al. 1997) even when the inverse 3-D Var is difficult to obtain as is the case with full physics models.

5 Hessian information in optimization theory

Hessian information is crucial in many aspects of both constrained and unconstrained minimization. All minimization methods start by assuming a quadratic model in the vicinity of the minimum of a multivariate minimization problem.

For the problem

$$\min_{\mathbf{X}\in\mathcal{R}^n} F(\mathbf{X}) \tag{87}$$

We require in the multivariate case that

$$\|\mathbf{g}(\mathbf{X}^*)\| = 0 \tag{88}$$

where $\mathbf{X}^*$ is a stationary point and in order to obtain sufficient conditions for the existence of the minimum of the multivariate unconstrained minimization problem, we must require that the Hessian be positive definite.

5.1 Spectrum of the Hessian and rate of convergence of unconstrained minimization

The eigenvalues of the Hessian matrix predict the behavior and convergence rate for unconstrained minimization. To show this, let us assume that a multivariate nonlinear function $F(\mathbf{X})$ satisfies for some $\mathbf{X}$,

$$|F(\mathbf{X}) - F(\mathbf{X}^*)| \leq \epsilon_A \tag{89}$$

and we define this as an acceptable solution.

If F is twice continuously differentiable, i.e., $F \in C^2$, and if $\mathbf{X}^*$ is a strong minimum then

$$\mathbf{g}(\mathbf{X}^*) = 0 \tag{90}$$

where $\mathbf{g}(\mathbf{X}^*)$ is the gradient vector of $F(\mathbf{X})$ and $G(\mathbf{X}^*)$ its Hessian is positive-definite, i.e., for $\mathbf{X}^*$

$$\mathbf{X}^{*T} G \mathbf{X}^* > 0 \tag{91}$$

Let us expand F in a Taylor series about $\mathbf{X}^*$

$$F(\mathbf{X}) = F(\mathbf{X}^* + h\mathbf{p}) = F(\mathbf{X}^*) + \frac{1}{2} h^2 \mathbf{p}^T G(\mathbf{X}^*)\mathbf{p} + O(h^2) \qquad (\text{since } \mathbf{g}(\mathbf{X}^*) = 0) \tag{92}$$

where

$$\|\mathbf{p}\| = 1 \qquad \text{and} \qquad h = \|\mathbf{X} - \mathbf{X}^*\| \tag{93}$$

For any acceptable solution we obtain

$$h^2 = \|\mathbf{X} - \mathbf{X}^*\|^2 \approx \frac{2\epsilon_A}{\mathbf{p}^T G(\mathbf{X}^*)\mathbf{p}} \tag{94}$$

substantially affects size of $\|\mathbf{X} - \mathbf{X}^*\|$, i.e., rate of convergence of the unconstrained minimization (Gill 1981).

If $G(\mathbf{X}^*)$ is ill-conditioned, the error in $\mathbf{X}$ will vary with the direction of the perturbation p.

If $\mathbf{p}$ is a linear combination of eigenvectors of $G(\mathbf{X}^*)$ corresponding to the largest eigenvalues, the size of $\|\mathbf{X} - \mathbf{X}^*\|$ will be relatively small, while if, on the other hand p is a linear combination of eigenvalues of $G(\mathbf{X}^*)$ corresponding to the smallest eigenvalues, the size of $\|\mathbf{X} - \mathbf{X}^*\|$ will be relatively large, i.e., slow convergence.

5.2 Role of the Hessian in constrained minimization

The Hessian information plays a very important role in constrained optimization as well.

We shall deal here with optimality conditions where again Taylor series approximations are used to analyze the behavior of the objective function F and constraints h_i about a local constrained minimizer $\mathbf{X}^*$.

We shall consider first optimal conditions for linear equality constraints.

1. The problem is cast as

$$\min_{\mathbf{X}\in R^n} F(\mathbf{X}), \qquad \text{subject to} \;\; A\mathbf{X} = \mathbf{b} \tag{95}$$

where A is an $m \times n$ matrix, $m \leq n$.

We assume F is twice continuously differentiable and that rows of A are independent, i.e., A has full row rank.

Let Z be the null space of A of dimension $n \times r$ with $r \geq n - m$. Then

$$AZ = 0 \tag{96}$$

Then the constrained minimization problem in $\mathbf{X}$ is equivalent to the unconstrained problem

$$\min_{\mathbf{v}\in\mathcal{R}^r} \Phi(\mathbf{v}) = F(\mathbf{X} + Z\mathbf{v}) \tag{97}$$

where $\mathbf{X}$ is a feasible point (Gill 1981, Nash and Sofer, 1996). The function Φ is the restriction of f onto the feasible region, called the reduced function.

If Z is a basis matrix for the null space of A, then Φ is a function of $n - m$ variables.

Optimality conditions involve derivatives of the reduced function. If $\mathbf{X} = \bar{\mathbf{X}} + Z\mathbf{v}$

$$\begin{aligned} \nabla\Phi(\mathbf{v}) &= Z^T\nabla F(\bar{\mathbf{X}} + Z\mathbf{v}) = Z^T\nabla F(\mathbf{X}) \\ \nabla^2\Phi(\mathbf{v}) &= Z^T\nabla^2 F(\bar{\mathbf{X}} + Z\mathbf{v})Z = Z^T\nabla^2 f(\mathbf{X})Z \end{aligned} \tag{98}$$

The vector

$$\nabla\Phi(\mathbf{v}) = Z^T\nabla f(\mathbf{X}) \tag{99}$$

is called the reduced gradient of f at $\mathbf{X}$.

Similarly the matrix

$$\nabla^2\Phi(\mathbf{v}) = Z^T\nabla^2 f(\mathbf{X})Z \tag{100}$$

is called the reduced or projected Hessian matrix.

The reduced gradient and Hessian matrix are the gradient and Hessian of the restriction of f onto the feasible region evaluated at $\mathbf{X}$.

If $\mathbf{X}^*$ is a local solution of the constrained problem then

$$\mathbf{X}^* = \bar{\mathbf{X}} + Z\mathbf{v}^* \qquad \text{for some } \mathbf{v}^* \tag{101}$$

and $\mathbf{v}^*$ is the local minimizer of Φ. Hence we can write

$$\nabla\Phi(\mathbf{v}^*) = 0 \tag{102}$$

and $\nabla^2\Phi(\mathbf{v}^*)$ is positive semi-definite.

Necessary conditions for a local minimizer. If $\mathbf{X}^*$ is a local minimizer of F and Z is the null-space matrix for a

$$Z^T\nabla F(\mathbf{X}^*) = 0 \tag{103}$$

and $Z^T\nabla^2 F(\mathbf{X}^*)Z$ is positive semi-definite. That is the reduced gradient is zero and the reduced Hessian matrix is positive semi-definite (the second order derivative information is used to distinguish local minimizers from other stationary points.)

The second order condition is equivalent to the condition

$$\mathbf{v}^T Z^T\nabla^2 F(\mathbf{X}^*)Z\mathbf{v} \geq 0 \qquad \text{for all } \mathbf{v} \tag{104}$$

Noting that $\mathbf{p} = Z\mathbf{v}$ is a null space vector, we can rewrite (104) as

$$\mathbf{p}^T\nabla^2 F(\mathbf{X}^*)\mathbf{p} \geq 0 \qquad \text{for all } \mathbf{p} \in \mathbf{v}(A) \tag{105}$$

i.e., the Hessian matrix at $\mathbf{X}^*$ must be positive semi-definite on the null space of A.

5.3 Example(Nash and Sofer, 1996)

Consider

$$\min f(\mathbf{X}) = x_1^2 - 2x_1 + x_2^2 - x_3^2 + 4x_3 \tag{106}$$

$$s.t. \qquad x_1 - x_2 + 2x_3 = 2 \tag{107}$$

$$\mathbf{X} = (x_1, x_2, x_3)^T$$

$$\nabla f(\mathbf{X}) = (2x_1 - 2, 2x_2, -2x_3 + 4)^T \tag{108}$$

Consider feasible point $\mathbf{X}^* = (2.5, -1.5, -1)^T$,

$$\nabla f(\mathbf{X}^*) = (3, -3, 6)^T \tag{109}$$

Selecting

$$Z = \begin{pmatrix} 1 & -2 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{110}$$

as the null space of matrix, $A = (1, -1, 2)^T$, one finds

$$Z^T \nabla f(\mathbf{X}^*) = \begin{pmatrix} 1 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ -3 \\ 6 \end{pmatrix} = (0, 0)^T \tag{111}$$

Reduced gradient vanishes and first-order necessary conditions are satisfied, checking the reduced Hessian matrix we find

$$Z^T \nabla^2 f(\mathbf{X}^*) Z = \begin{pmatrix} 1 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -2 \end{pmatrix} \begin{pmatrix} 1 & -2 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 4 & -4 \\ -4 & 6 \end{pmatrix} \tag{112}$$

i.e., reduced Hessian matrix is partial differential at $\mathbf{X}^*$. However we notice that $\nabla^2 f(\mathbf{X}^*)$ itself is not positive definite.

5.4 Optimality conditions for nonlinear constraints

Consider the problem

$$\min f(\mathbf{X}) \text{ subject to } \; g_i(\mathbf{X}) = 0, \quad i = 1, 2, ..., m \tag{113}$$

Optimality conditions are expressed in terms of the Lagrangian function

$$L(\mathbf{X}, \lambda) = f(\mathbf{X}) - \sum_{i=1}^{m} \lambda_i g_i(x) = f(\mathbf{X}) \lambda^T \mathbf{g}(\mathbf{X}) \tag{114}$$

λ is a vector of Lagrangian multiplier, $\mathbf{g}(\mathbf{X})$ is a vector of nonlinear constraint functions (g_i).

5.5 Necessary conditions for nonlinear equality constraints

Let $\mathbf{X}^*$ be a local minimizer of f subject to the constraints $g(\mathbf{X}) = 0$. Let $Z(\mathbf{X}^*)$ be the null-space matrix for the Jacobian matrix $\nabla g(\mathbf{X}^*)^T$. If $\mathbf{X}^*$ is a regular point of the constraints, there exists a vector of Lagrangian multiplier $\mathbf{X}^*$ such that $\nabla_{\mathbf{X}} L(\mathbf{X}^*, \lambda^*) Z(\mathbf{X}^*)$ or equivalently $Z(\mathbf{X}^*)^T \nabla f(\mathbf{X}^*) = 0$, $Z(\mathbf{X}^*)^T \nabla^2_{\mathbf{XX}} L(\mathbf{X}^*, \lambda^*) Z(\mathbf{X}^*)$ is positive definite.

The second order optimality conditions are based on the reduced Hessian

$$Z(\mathbf{X}^*)^T \nabla^2_{\mathbf{XX}} L(\mathbf{X}^*, \lambda^*) Z(\mathbf{X}^*) \tag{115}$$

and involve the Hessian of the Lagrangian L.

5.6 Use of second order information for optimization algorithms

Second order information can be used to improve performance of efficient large-scale minimization algorithms.

It is known in optimization theory that Newton and Truncated-Newton minimization algorithms display a quadratic rate of convergence and thus can sizably reduce the number of iterations required to achieve a prescribed convergence criterion in 4-D Var where large scale unconstrained optimization is required.

It was however a widespread belief in the geosciences community that obtaining the Hessian of the cost functional(i.e., second order derivative information) is very difficult and very expensive to compute (Santosa and Symes 1988). It become clear that in Truncated-Newton algorithm (Wang 1995) one never needs to compute the Hessian operator explicitly but rather one needs to know its action on a given vector, and therefore the huge matrix representing the Hessian does not need to be stored.

5.7 Application of second-order-adjoint technique to obtain exact Hessian/vector product

We will exemplify this application by considering a Truncated-Newton algorithm for large-scale minimization.

Description of Truncated-Newton methods.

Truncated-Newton methods are used to solve the problem

$$\min f(\mathbf{X}), \qquad \mathbf{X} = (x_1, x_2, ..., x_n)^T \tag{116}$$

They are a compromise on Newton method whereby they compute a search direction by finding an approximate solution to the Newton's equations

$$\nabla^2 f(\mathbf{X}_k)\mathbf{p} \approx -\nabla f(\mathbf{X}_k) \tag{117}$$

using a conjugate-gradient iterative method, we note here that Newton equations are a linear system of the form

$$A\mathbf{X} = \mathbf{b} \tag{118}$$

where

$$\begin{aligned} A &= \nabla^2 f(\mathbf{X}_k) \\ \mathbf{b} &= -\nabla f(\mathbf{X}_k) \end{aligned} \tag{119}$$

The conjugate gradient method is "truncated" before the exact solution to the Newton equations has been found. The C-G method computes the search direction, and requires storage of a few vectors.

The only obstacle for using minimization is the requirement that it computes matrix-vector products of the lines

$$A\mathbf{V} = \nabla^2 f(\mathbf{X}_k)\mathbf{v} \tag{120}$$

for arbitrary vectors $\mathbf{v}$.

One way to bypass the storage difficulty is to approximate the matrix-vector products using values of the gradient in such a way that the Hessian matrix need not be computed or stored.

Using Taylor series

$$\nabla f(\mathbf{X}_k + h\mathbf{v}) = \nabla f(\mathbf{X}_k) + h\nabla^2 f(\mathbf{X}_k)\mathbf{v} + O(h^2) \tag{121}$$

we obtain

$$\nabla^2 f(\mathbf{X}_k)\mathbf{v} = \lim_{h\to 0} \frac{\nabla f(\mathbf{X}_k + h\mathbf{v}) - \nabla f(\mathbf{X}_k)}{h} \tag{122}$$

i.e., we approximate matrix vector product

$$\nabla^2 f(\mathbf{X}_k)\mathbf{v} \approx \frac{\nabla f(\mathbf{X}_k + h\mathbf{v}) - \nabla f(\mathbf{X}_k)}{h} \tag{123}$$

for some small values of h.

The task of choosing an adequate h is an arduous one(see Nash and Sofer 1996, Chapter 11.4.1 and references therein).

For in-depth descriptions of the truncated-Newton (also referred to as the Hessian-free) method see Nash (1982), Nash (1984a,b,c,d), Nash (1985), Nash and Sofer (1989a,1989b) as well as Schlick and Fogelson (1992a,1992b). A comparison of Limited Memory quasi-Newton and Truncated -Newton methods is provided by Nash and Nocedal (1991), while a comprehensive well-written survey of truncated-Newton methods is presented in Nash (2000). Due to availability of second-order adjoint information, a description of which is provided in Wang (1992, 1993),Wang et al.(1995, 1997,1998), one can obtain a better approximation to the Newton line search allowing a major speed-up of the convergence rate.

A comparison between limited memory quasi-Newton and truncated-Newton methods applied to a meteorological problem is described in depth by Zou et al.(1990, 1993).

5.8 The adjoint Truncated-Newton method

This algorithm(ATN) differs from usual Truncated-Newton algorithm only in the Hessian-vector product calculation solving Newton equations at k-th iteration

$$G_k \mathbf{p}_k = -\mathbf{g}_k \tag{124}$$

where $G_k = \nabla^2 f(\mathbf{X}_k)$ is the Hessian of the cost function $f(\mathbf{X}_k)$, $\mathbf{p}_k$ is the linear search direction and

$$\mathbf{g}_k = \nabla f(\mathbf{X}_k) \tag{125}$$

Here the second order adjoint produces an exact Hessian-vector product. The main steps of the adjoint truncated Newton algorithm are:

1. Choose $\mathbf{X}_0$ initial guess to the minimizer $\mathbf{X}^k$, set iteration counter to zero.
2. Test $\mathbf{X}_k$ for convergence. Check if following convergence criterion is satisfied:

$$\|\mathbf{g}_k\| < 10^{-5}\|\mathbf{g}_0\| \tag{126}$$

If the criterion is satisfied then stop. Otherwise continue.

3. Solve Newton equations approximately using a preconditioned truncated C-G algorithm where exact Hessian-vector product is obtained using the second order adjoint model.

4. Set $k = k + 1$, update

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \alpha_k \mathbf{p}_k \tag{127}$$

where α_k is the step-size obtained by conducting a line-search using Davidon's cubic interpolation method. Go to step 2.

5.9 A method for estimating the Hessian matrix

Consider a cost function measuring the misfit between forecast model solution and available observations distributed in space and time.

$$\begin{aligned} J(x(t_0)) &= \frac{1}{2}\{B(x(t_0)) - x^{obs}(t_0)\}^T \cdot W(t_0)\{B(x(t_0)) - x^{obs}(t_0)\} \\ &+ \frac{1}{2} < B\{F(x(t_0))\} - x^{obs}(t_n) >^T \cdot W(t_n) < B(F(x(t_0)) - x^{obs}(t_n) > \end{aligned} \tag{128}$$

where B is an observation operator, $x(t_r)$ the vector of model control variables, $x^{obs}(t_r)$ the vector of observational data at time $t = t_r$, $w(t_r)$ the inverse of observation covariance matrix.

$$F = \prod_{n=1}^{N} F_n \tag{129}$$

the operator of model integration from time $t = t_0$ to $t = t_N$. At the minimum of above expression the gradient of the cost function ∇J vanishes.

If we introduce random variables $\eta(t_0)$ and $\eta(t_N)$ with zero expectations and whose covariances are the diagonal elements of $W^{-1}(t_0)$ and $W^{-1}(t_N)$ respectively to the observations

$$x_1^{obs}(t_0) = x^{obs}(t_0) + \eta(t_0), \tag{130}$$

$$x_1^{obs}(t_N) = x^{obs}(t_N) + \eta(t_N). \tag{131}$$

then ∇J at x_{min} is random variable and we get

$$< \nabla J, \nabla^T J >= J'' \tag{132}$$

where the angle in brackets stands for the mathematical expectation and J'' is the Hessian matrix. We can see that we obtain an outer vector product expression, which is rank-one matrix.

For each realization i of $x_1^{obs}(t_0)$ and $x_1^{obs}(t_N)$ we can calculate ∇J^i at x_{min} and after p such realizations we obtain at most a rank p approximation of the Hessian of the cost function (Yang et al.1996, Rabier and Courtier 1992, Courtier et al. 1994)

$$H \approx J'' \approx J_p'' = \frac{1}{p} \sum_{i=1}^{P} \nabla J^i \cdot \nabla (J^i)^T \tag{133}$$

This approach is analogous to Quasi-Newton methods where symmetric rank 1 or rank-two update as are collected to update approximation of the Hessian or the inverse of the Hessian matrix as the minimization proceeds. As shown by Yang et al. (1996) use of the approximate J_p'' as preconditioner is extremely efficient. Forsythe and Strauss (1955) have already shown that using the diagonal of the Hessian is optimal amongst all diagonal preconditioning methods.

6 Second Order Adjoint via Automatic Differentiation

There is an increased interest in obtaining the second order adjoint via Automatic Differentiation (A.D.).

Research work has been carried out both in the recent version of the Fortran TAMC AD package designed by Giering and Kaminski(1998a) allowing for both the calculation of Hessian/vector products as well as for the more computationally expensive derivation of the full Hessian with respect to the control variables. See also Giering and Kaminski 1998b.

Comparable CPU times to these required by hand coding were reported.

The importance of the Hessian/vector products derived by A.D. is particularly important in minimization where there is often interest not only in the first but rather in the second derivatives of the cost functional which convey crucial information.

Griewank (2000) in his new book estimated the computational complexity of implementing second order adjoints in a thorough manner.

He found that for calculating Hessian/vector products an effort leading to a run-time ratio of about a factor of 13 was required.

The calculation of the ratio between the effort required to obtain Hessian/ vector products against that required to calculate the gradient of the cost was found to be a factor between 2–3 only.

Exploiting sparsity for A.D. calculation of the second order adjoint Griewank (2000) shows that economy can be realized when the graph symmetry allows the AD computed Hessian to assume the form:

$$\nabla^2 f = \dot{Z} S \dot{Z} \in \mathcal{R}^{n \times n} \tag{134}$$

which leads to a dyadic representation first put forward in a paper by Jackson and McCormick (1988).

Here:

$$Z \equiv (\dot{V}_i^T)_{i=1 \cdots n}^{l-m} \in \mathcal{R}^{(l-m+n) \times n}, \tag{135}$$

$$S \in \mathcal{R}^{(l-m+n) \times (l-m+n)}, \tag{136}$$

where

$$\dot{V}_i^T = \nabla \times v_i, \tag{137}$$

v_i which are intermediates with respect to independents $x \in \mathcal{R}^n$.

7 Computational Complexity of A.D. calculation of the 2-nd order adjoint

Griewank (2000) starts by working out a representation of the complexity measure as a task consisting of moves, adds, multiplications and nonlinear operations , thus obtaining a representation of work (task) as:

$$Work(task) = \begin{pmatrix} moves \\ adds \\ mults \\ nlops \end{pmatrix} = \begin{pmatrix} \text{no of fetches and stores} \\ \text{no. of adds and subtracts} \\ \text{no. of multiplications} \\ \text{no of nonlinear operations} \end{pmatrix}$$

Then runtime can be written as:

$$TIME(task(F)) = \mathbf{w}^T work(task(F))$$

Here $\mathbf{w}$ is a vector of $dim(work)$ of positive weights which depend on the computing system and represent the number of clock cycles needed for fetching and/or storing data items, multiplication, addition, and finally for taking into account nonlinear operations.

Usually the vector $\mathbf{w}^T$ assumes the form

$$\mathbf{w}^T = (\mu, 1, \pi, \nu) \tag{138}$$

and for most computing platforms $\mu \geq \max(1, \pi/2)$, $\pi \leq 1$ and $\nu \leq 2\pi$. For example, this assumption implies that a memory access (μ) is at least as slow as an addition or half a multiplication (π). Griewank (2000) derives the computational complexity of the tangent model (directional derivative) w_{tang}, gradient (first order adjoint) w_{grad}, and second order adjoint w_{SOAD} normalized by the complexity of the model evaluation as

$$w_{tang} = \max\{\frac{2\mu}{\mu}, \frac{6\mu+2}{3\mu+1}, \frac{6\mu+1+3\pi}{3\mu+\pi}, \frac{4\mu+\pi+2\nu}{2\mu+\nu}\} \in [2, \frac{5}{2}] \tag{139}$$

$$w_{grad} = \max\{\frac{2\mu}{\mu}, \frac{9\mu+3}{3\mu+1}, \frac{11\mu+2+3\pi}{3\mu+\pi}, \frac{7\mu+1+\pi+2\nu}{2\mu+\nu}\} \in [3, 4] \tag{140}$$

$$w_{SOAD} = \max\{\frac{4+\mu}{\mu}, \frac{18\mu+6}{3\mu+1}, \frac{22\mu+7+9\pi}{3\mu+\pi}, \frac{\mu+3+5\pi+3\nu}{2\mu+\nu}\} \in [7,10] \tag{141}$$

As mentioned by Nocedal and Wright (1999) automatic differentiation has been increasingly using more sophisticated techniques that allow when used in reverse mode to calculate either full Hessians or Hessian/vector products . However the automatic differentiation technique should not be regarded as a substitute for the user to think that this is a fail-safe product and each derivative calculation obtained with A.D. should be carefully assessed.

Gay (1996) has shown how to use partial separability of the Hessian in A.D. while Powell and Toint (1979) and Coleman and More (1984) along with Coleman and Cai (1986) have shown how to estimate sparse Hessian using either graph coloring techniques or other highly effective schemes.

Software for the estimation of sparse Hessians is available in the work of Coleman, Garbow and More (1985a, 1985b).

Averbukh et al. (1994) supplemented the work of More et al. (1981) (ACM Trans. Math. Softw. 7, 14-41, 136-140, 1981) which provides function and gradient subroutines of 18 test functions for multivariate minimization. Their supplementary Hessian segments enable users to test optimization software that requires second derivative information.

8 Use of Hessian of cost functional to estimate error covariance matrices

A relationship exists between the inverse Hessian matrix and the analysis error covariance matrix of either 3-D VAR or 4-D VAR (See Thacker 1989, Rabier and Courtier 1992, Yang et al. 1996, LeDimet et al. 1997).

If $\mathbf{x}$ represents all variables being optimized and $\mathbf{x}^*$ is the optimal value sought , then in the vicinity of the minimum we have:

$$J = Cte + (x - x^*)^T H((x - x^*) + H.O.T \tag{142}$$

where $H.O.T.$ means higher order terms.

Alternatively using standard formulation of 3-D and 4-D VAR (See Ide 1997 , Le Dimet and Talagrand 1986), we have then:

$$\nabla^2 J(\mathbf{x}_0) = B^{-1} + (H')^T(x_0)R^{-1}H'(x_0) + H.O.T., \tag{143}$$

where $\mathbf{B}$ is the covariance matrix of the background error, $\mathbf{R}$ is the observation covariance matrix, $\mathbf{H}'$ is the nonlinear observation operator.

Hence the analysis error covariance matrix is:

$$B^{-1} + (H')^T(x_0)R^{-1}H'(x_0) \tag{144}$$

and can be used provided we approximate the inverse Hessian of the cost in the vicinity of the minimum of the cost functional.

A requirement is that the background error and the observation error are uncorrelated (Rabier and Courtier 1992, Fisher and Courtier 1995).

9 Hessian Singular Vectors (HSV)

Computing HSV's uses the full Hessian of the cost function in the variational data assimilation which can be viewed as an approximation of the inverse of the analysis error covariance matrix and it is used at initial time to define a norm.

The total energy norm is still used at optimization time. See work by Barkmeijer et al. (1998, 1999).

The HSV's are consistent with the 3-D VAR estimates of the analysis error statistics.

In practice one never knows the full 3-D VAR Hessian in its matrix form and a generalized eigenvalue problem is solved as we will describe below.

The HSV's are also used in a method first proposed by Courtier (1993) and tested by Rabier et al. (1997) for the development of a simplified Kalman filter fully described by Fisher (1998) and compared with a low resolution explicit extended Kalman filter by Ehrendorfer and Bouttier (1998).

Let M be the propagator of the Tangent linear model, P a projection operator setting a vector to zero outside a given domain.

Consider positive-definite and symmetric operators including a norm at initial and optimization time respectively.

Then the SV's defined by

$$\frac{< P\epsilon(t), EP\epsilon(t) >}{< \epsilon(t_0), C\epsilon(t_0) >} \tag{145}$$

under an Euclidean norm are solution of generalization eigenvalue problem.

$$M^* P^* EPMx = \lambda Cx. \tag{146}$$

In HSV, the operator C is equal to the Hessian of the 3-D Var cost function.

As suggested by Barkmeijer et al. (1998), one can solve (146) by a generalized algorithm (Davidson 1975). See also Sleijpen and Van der Vorst (1996). Using

$$C \equiv \nabla^2 J = B^{-1} + H^T R^{-1} H \tag{147}$$

and carrying out a coordinate transformation

$$x = L^{-1}x, \quad LL^{-1} = B. \tag{148}$$

Then we have a transformed operator

$$(L^{-1})^T CL \tag{149}$$

and the Hessian becomes equal to the sum of identity and a matrix with rank less or equal to the dimensions of the vector of observations Fisher and Courtier (1995).

Versee (1999) proposes to take advantage of this form of the appropriate Hessian in order to obtain approximations of the inverse analysis error covariance matrix, using the limited inverse BFGS minimization algorithm.

Let H be $(\nabla^2 J)^{-1}$ the inverse Hessian and H^+ the updated version of the inverse Hessian.

$$S = x^{n+1} - x^n \tag{150}$$

$$y = g^{n+1} - g^1 \tag{151}$$

One has the formula

$$H^+ = U(H, y, s) = (I - \frac{s \otimes y}{< y, s >})\frac{s \otimes s}{< y, s >} \tag{152}$$

$<,>$ is a scalar product and $\otimes$ is the outer product.

These methods are useful when the second order adjoint method is not available due to either memory or CPU limitations.

10 Numerical experiments: Application of AD Hessian/vector products to the Truncated Newton algorithm

For the numerical experiments we consider the truncated Newton algorithm to minimize the cost function (59) associated to the SWE model (56) - (58). The spatial domain considered is a 6000 km $\times$ 4400 km channel with a uniform 21×21 spatial grid, such that the dimension of the initial condition vector $(u, v, \phi)^t$ is 1083, and the Hessian of the cost function is a 1083×1083 matrix.

The initial conditions are those of Grammeltvedt (1969).As for the boundary conditions, on the southern and northern boundaries the normal velocity components are set to zero, while periodic boundary conditions are assumed in the west-east direction. Integration is performed with a time increment $\Delta t = 600s$ and the length of the assimilation window is ten hours. Data assimilation is implemented in a twin experiments framework such that the value of the cost function at the minimum point must be zero. As the set of control parameters we consider the initial conditions which are perturbed with random values chosen from an uniform distribution.

The second order adjoint model was generated using the tangent linear and adjoint model compiler TAMC (Giering and Kaminski 1998a). The correctness of the adjoint generated routines was checked using the small perturbations technique. Assuming that the cost function $f(\mathbf{X})$ is evaluated by the subroutine $model(f, \mathbf{X})$, computation of the Hessian/vector products $H(\mathbf{X})\mathbf{u}$ via automatic differentiation is performed in two steps: first the reverse (adjoint) mode is applied to generate the adjoint model

$$admodel(\mathbf{X}, f, \mathbf{adX}, adf)$$

Next, the tangent (forward) mode is used to generate the SOA model

$$g\_admodel(\mathbf{X}, f, \mathbf{adX}, adf, \mathbf{g\_X}, \mathbf{g\_adX})$$

We initialize the tangent state vector $\mathbf{g\_X} \equiv \mathbf{u}$, the adjoint state vector $\mathbf{adX} \equiv 0$, the second order adjoint $\mathbf{g\_adX} \equiv 0$, and $adf = 1$. On exit the computed gradient value is $\nabla f(\mathbf{X}) = \mathbf{adX}$ and the value of the Hessian/vector product $H(\mathbf{X})\mathbf{u}$ is returned in $\mathbf{g\_adX}$. The performance of the minimization process using AD SOA is analyzed versus an approximate Hessian/vector product computation given by (102), with a hand code adjoint model implementation. The absolute and relative differences between the computed Hessian/vector product at the first iteration (initial guess state) are shown in Figure 1 for the first 100 components. The first order finite difference method (FD) provides in average an accuracy of 2-3 significant digits. The optimization process using FD stops after 28 iterations when the line search fails to find a descending direction, whereas for the SOA method a relative reduction in the cost function up to the machine precision is reached at iteration 29. The evolution of the normalized cost function and gradient norm are presented in Figure 2 and Figure 3 respectively.

The computational cost is of same order of magnitude for both the finite-difference approach and the exact second-order adjoint approach. The second-order adjoint approach requires integrating the original nonlinear model and its tangent linear model(TLM) forward in time and integration of first order adjoint model and second order adjoint model backward in time once. The average ratio of the CPU time required to compute the gradient of the cost function to the CPU time of evaluating the cost function was $cpu(\nabla f)/cpu(f) \approx 3.7$. If we assume that the value of the gradient $\nabla f(\mathbf{X})$ in (123) is already available (previously computed in the minimization algorithm), to evaluate the Hessian×vector product using the FD method only one additional gradient evaluation $\nabla f(\mathbf{X}+hv)$ is needed in (123). In this case, we have then an average ratio to compute the Hessian×vector product $cpu(Hu)_{FD}/cpu(f) \approx 3.7$. Using the SOA method to compute the exact Hessian×vector product we obtained an average $cpu(Hu)_{SOA}/cpu(f) \approx 9.4$, in agreement with the estimate (141). We notice that in addition to the Hessian×vector product the AD SOA implementation provides also the value of the gradient of the cost function. The average ratio $cpu(Hu)_{SOA}/cpu(\nabla f) \approx 2.5$ we obtained is also in agreement with the CPU estimates derived in Section 7.

10.1 Numerical calculation of Hessian eigenvalues

Iterative methods and the SOA model may be combined to obtain information about the spectrum of the Hessian matrix of the cost function. In this application we used the ARPACK package (Lehoucq et al. 1998) to compute five of the largest and smallest eigenvalues of the Hessian matrix. The method used is the implicitly restarted Arnoldi method (IRAM) which reduces to the implicitly restarted Lanczos method (IRLM) since H is symmetric. For our application, only the action of the Hessian matrix on a vector is needed and we provide this routine using the SOA model. The condition number is evaluated as

$$k(H) = \frac{\lambda_{max}}{\lambda_{min}} \tag{153}$$

The computed Ritz values and the relative residuals are included in Table 1 for the Hessian evaluated at the initial guess point, and in Table 2 for the Hessian evaluated at the optimal point $\mathbf{X}^{\star}$. For our test example the eigenvalues of the Hessian are positive, such that the Hessian is positive definite and the existence of a minimum point is assured. The condition number of the Hessian is of order $k(H) \sim 10^4$ which explains the slow convergence of the minimization process.

Use of Hessian of a cost function eigenvalue information in regularization of ill-posed problems was illustrated by Alekseev and Navon (2000a, 2000b). The application consisted of wavelet regularization approach for dealing with an ill-posed problem of adjoint parameter estimation applied to estimating inflow parameters from down-flow data in an inverse convection case applied to the two-dimensional parabolized Navier- Stokes equations. The wavelet method provided a decomposition into two subspaces, by identifying both a well-posed as well as an ill- posed subspace, the scale of which was determined by finding the minimal eigenvalues of the Hessian of a cost functional measuring the lack of fit between model prediction and observed parameters. The control space is transformed into a wavelet space. The Hessian of the cost was obtained either by a discrete differentiation of the gradients of the cost derived from the first-order adjoint or by using the full second-order adjoint. The minimum eigenvalues of the Hessian are obtained either by employing a shifted iteration method Zou et al.(1992) or by using the Rayleigh quotient. The numerical results obtained illustrated the usefulness and applicability of this algorithm if the Hessian minimal eigenvalue is greater or equal to the square of the data error dispersion, in which case the problem can be considered as well-posed (i.e., regularized). If the regularization fails, i.e., the minimal Hessian eigenvalue is less than the square of the data error dispersion of the problem, the following wavelet scale should be neglected, followed by another algorithm iteration.

11 Summary and Conclusions

The recent development of variational methods in operational meteorological centers (ECMWF, Meteo-France) has demonstrated the strong potential of these methods.

Variational techniques require the development of powerful tools such as the adjoint

model, which are useful for the adjustment of the inputs of the model (initial and/or boundary conditions). From the mathematical point of view the first order adjoint will provide only necessary conditions for an optimal solution. The second order analysis goes one step further and provides an information, which is essential for many applications:
i) sensitivity analysis should be derived from a second order analysis i.e. from the derivation of the optimality system. This is made crystal clear when sensitivity with respect to observations is required. In the analysis observations appear only as a forcing term in the adjoint model, therefore in order to estimate the impact of observations this is the system that should be derived.
ii) second order information will improve the convergence of the optimization methods, which are the basic algorithmic component of variational analysis.
iii) the second order system permits to estimate the covariances of the fields. This information is essential for the estimation of the impact of errors on the prediction.

The computational cost to be paid in order to obtain the second order adjoint system is twofold:

i) We have to consider the computational cost for the derivation of the SOA. It has been seen that we can get it directly from the linear tangent model and from the adjoint model. Only the right hand sides should be modified.
ii) Computing the second order information. Basically the first order information has the same dimension as the input of the model. Let n be this dimension. The second order information will be represented by $n \times n$ matrix. For operational models the computation of the full Hessian matrix is prohibitive, nevertheless it is possible to extract the most useful information (eigenvalues and eigenvectors, spectrum, condition number, $\cdots$) at a reasonable computational cost.

The numerical results obtained illustrate the ease with which present day automatic differentiation packages allow to obtain second order adjoint model as well as Hessian/vector product.They also confirm numerically the CPU estimates for computational complexity as derived in Section 7 (See also Griewank (2000)).

Numerical calculation of the leading eigenvalues of the Hessian along with its smallest eigenvalues yields results similar to those obtained by Wang et al. (1998) and allow valuable insight into the Hessian spectrum, thus allowing us to deduct the important information related to condition number of the Hessian, hence to the expected rate of convergence of minimization algorithms.

With the advent of ever more powerful computers ,the use of second order information in data assimilation will be within realistic reach for 3-D models and is expected to become more prevalent.

The purpose of this paper was to demonstrate the importance of new developments in second order analysis: many directions of research remain open in this domain.

Acknowledgements

The second author would like to acknowledge the support from the NSF grant number ATM-9731472 managed by Dr. Pamela Stephens whom we would like to thank for her support.

References

[1] Abate,J., C. Bischof, A. Carle, and L. Roh,1997:Algorithms and Design for a Second-Order Automatic Differentiation Module.in *Proc. Int. Symposium on Symbolic and Algebraic Computing (ISSAC) '97*,Association of Computing Machinery, New York,149-155.

[2] Alekseev, K.A. and Navon I.M., 2000a: The Analysis of an Ill-Posed Problem Using Multiscale Resolution and Second Order Adjoint Techniques. *Computer Methods in Applied Mechanics and Engineering*, **In press**.

[3] Alekseev, K.A. and Navon I.M., 2000b:On Estimation of Temperature Uncertainty Using the Second Order Adjoint Problem. *Submitted for publication to International Journal of Computational Fluid Dynamics*

[4] Arian,E. and S. Ta'asan,1999:Analysis of the Hessian for aerodynamic optimization: inviscid flow.*Computers & Fluids*,**28: (7)**, 853-877.

[5] Averbukh V.Z., Figueroa S. and Schlick T.,1994: Remark on Algorithm-566. *ACM Trans. Math Software* 20: (3) 282-285 .

[6] Barkmeijer, J., Buizza R. and Palmer T.N.,1999: 3D-Var Hessian singular vectors and their potential use in the ECMWF Ensemble Prediction System. *Quart. J. Roy. Meteor. Soc.***125, Part B** 2333-2351. .

[7] Barkmeijer, J., Van Gijzen, M. and F. Bouttier ,1998: Singular vectors and estimates of the analysis-error covariance metric. *Quart. J. Roy. Meteor. Soc.* **124, Part A**, 1695-1713 .

[8] Bischof, C.H.,1995: Automatic Differentiation, Tangent Linear Models, and (Pseudo)Adjoints.*in Proceedings of the Workshop on High-Performance Computing in the Geosciences, Francois-Xavier Le Dimet, ed.*,NATO Advanced Science Institutes Series : C, Mathematical and Physical Sciences , 462, Kluwer Academic Publishers B.V.

[9] Burger,J., J. L. Brizaut and M. Pogu, 1992: Comparison of two methods for the calculation of the gradient and of the Hessian of the cost functions associated with differential systems. *Mathematics and Computers in Simulation*, **34**, 551-562.

[10] Bus, J. C. P., 1977: Convergence of Newton-like methods for solving systems of nonlinear equations. *Numerische Mathematik*, **27**, 271-281.

[11] Coleman, T.F.and Cai J.Y.,1986: The Cyclic Coloring Problem and Estimation of Sparse Hessian Matrices .*SIAM J. Algebra Discr.* **7 (2)**, 221-235 .

[12] Coleman, T.F., B.S. Garbow and J.J. More ,1985: Software for estimating Sparse Hessian Matrices. *ACM Trans Math Software* **11: (4)**, 363-377 .

[13] Coleman, T.F., B.S. Garbow and J.J. More ,1985a: Fortran Subroutines for estimating Sparse Hessian Matrices. *ACM Trans Math Software* **11: (4)**, 378-378.

[14] Coleman TF, More J.J.,1984: Estimation of sparse Hessian Matrices and Graph-coloring Problems. *Math Program*,**28: (3)**, 243-270 .

[15] Courtier, P. and O. Talagrand, 1987: Variational assimilation of meteorological observations with the adjoint equations Part 2. Numerical results. *Quart. J. Roy. Meteor. Soc.*, **113**, 1329-1347.

[16] Courtier, P., J.-N. Thepaut, and A. Hollingsworth, 1994: A strategy for operational implementation of 4D-Var, using an incremental approach. *Quart J. Roy. Meteor. Soc.*, **120**, 1367-1388.

[17] Courtier, P.,1993: Introduction to Numerical Weather prediction Data Assimilation Methods. Proc. ECMWF Seminar on Developments in the use of Satellite Data In Numerical Weather Prediction. 6-10 September 1993.

[18] Davidon, W. C., 1959: Variable metric method for minimization. A. E. C. *Research and Development Report*, ANL-5990 (Rev.).

[19] Davidson E.R.,1975: The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of a large real symmetric matrices. **J. Comput. Phys.**, **Vol 17**, 87-94.

[20] Dembo, R. S., S. C. Eisenstat and T. Steihaug, 1982: Inexact Newton methods. *SIAM Journal of Numerical analysis,* **19**, 400-408.

[21] Dembo, R. S. and T. Steihaug, 1983: Truncated-Newton algorithms for large-scale unconstrained optimization. *Math. Prog.*, **26**, 190,212.

[22] Dennis, J. and R. Schnabel, 1983: *Numerical methods for unconstrained Optimization and nonlinear equations.* EngleWood Cliffs, NJ: Prentice-Hall.

[23] Dennis, J., 1994: Personal communication. Dept. of Math., Rice University, Houston, TX.

[24] Dixon, L. C. W., 1991: Use of automatic differentiation for calculating Hessians and Newton steps, in *Automatic differentiation of algorithms: theory, implementation, and application,* Andreas Griewank and George F. Corliss (ed.), SIAM, Philadelphia, 115-125.

[25] Ehrendorfer, M. and F. Bouttier,1998: An explicit low-resolution extended Kalman filter: Implementation and preliminary experimentation. Sept 1998. ECMWF Technical Memorandum No 259.

[26] Errico, Ronald M., Tomislava Vukićević, and Kevin Raeder, 1993: Examination of the accuracy of a tangent linear model. *Tellus*, **45A**, 462-477.

[27] Fisher, M. and Courtier, P., 1995:Estimating the covariance matrices of analysis and forecast errors in variational data assimilation.ECMWF Tech Memorandum, No. 220, 28pp.

[28] Fisher, M.,1998: Development of a simplified Kalman filter.ECMWF Tech Memorandum, No. 260,16pp.

[29] Forsythe ,G.E. and E.G. Strauss,1955: On best conditioned matrices.*Proc. Amer. Math. Soc.*, **6**, 340-345.

[30] Gay , D.M.,1996: More AD of Nonlinear AMPL Models: Computing Hessian Information and Exploiting Partial Separability in Computational Differentiation: Techniques, Applications, and Tools Martin Berz, Christian Bischof, George Corliss, and Andreas Griewank, Editors, Proceedings in Applied Mathematics 89 .SIAM 1996 / xvi + 421 pages.

[31] Giering, R. and T. Kaminski ,1998a: Recipes for adjoint code construction *ACM Trans Math Software*,**24: (4)** ,437-474.

[32] Giering, R. and T. Kaminski ,1998b:Using TAMC to generate efficient adjoint code: Comparison of automatically generated code for evaluation of first and second order derivatives to hand written code from the Minpack-2 collection.*Extended abstracts of the Session Automatic Differentiation for adjoint code generation at the IMAC Conference on Applications of Computer Algebra, Prag, 1998.* Available at:http://puddle.mit.edu/ ralf/tamc/minpack/gierkam/node1.html.

[33] Gilbert, J. C., 1992: Automatic differentiation and iterative processes. *Optimization Methods and Software,* **1**, 13-21.

[34] Gill, P. E. and W. Murray, 1972: Quasi-Newton methods for unconstrained optimization. *J. Inst. Maths Applics*, **9**, 91-108.

[35] Gill, P. E. and W. Murray, 1974: *Numerical methods for unconstrained optimization.* Academic Press, London and New-York, 283pp.

[36] Gill, P. E. and W. Murray, 1979: Newton-type methods for unconstrained and linearly constrained optimization. *Math. Prog.*, **28**, 311-350.

[37] Gill P. E., Walter Murray and Margaret H. Wright, 1981 : Practical Optimization. Academic Press, London, 401pp.

[38] Gill, P. E., W. Murray, M. C. Saunders and M. H. Wright, 1983: Computing forward-difference intervals for numerical optimization. *SIAM J. Sci. Stat. Comput.*, **4**, 310-321.

[39] Grammeltvedt, A., 1969: A survey of finite-difference schemes for the primitive equations for a barotropic fluid. *Mon. Wea. Rev.*, **97**, 387-404.

[40] Griewank, A. and George F. Corliss., 1991: *Automatic differentiation of algorithms: theory, implementation, and application,* SIAM, Philadelphia, 353pp.

[41] Griewank, A.,1993: Some bounds on the complexity of gradients, Jacobians, and Hessians,*Complexity in Nonlinear Optimization (P.M. Pardalos, ed.)*, World Scientific Publishers, , pp. 128–161.

[42] Griewank, A.,2000: Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation. Frontiers in Applied Mathematics 19, SIAM , Philadelphia, 369pp.

[43] Hamming, R. W., 1973: *Numerical methods for scientists and engineers,* 2nd edition, McGraw-Hill and New York.

[44] Hovland ,P.,1995: Using ADIFOR 1.0 to Compute Hessians, CRPC-TR95540-S,Center for Research on Parallel Computation, Rice University, Houston, TX, 12 pp.

[45] Ide, K., P. Courtier, M. Ghil, and A. Lorenc, 1997: Unified notation for data assimilation: Operational sequential and variational. In Data Assimilation, Meteorology and Oceanography: Theory and Practice, *J. Meteorol. Soc Japan*, **75, 1B**, 71-79. Stiefel, 1952: Methods of conjugate gradients for 409-436.

[46] Hou, G J.-W. and J. Sheen, 1993: Numerical methods for second order shape sensitivity analysis with application to heat conduction problems. *Int. J. Num. Meth. in Engr.*, **36**, 417-435.

[47] Jackson R.H.F. and G.P. McCormic: Second order sensitivity analysis in factorable programming: Theory and Applications. Mathematical Programming Vol. 41, No. 1, 1-28.

[48] Kalnay E.,S.K. Park, Z-X. Pu and J Gao, 2000:Application of the quasi-inverse method to data assimilation. *Mon. Wea. Rev.*, **Vol. 128**, 864-875.

[49] Lacarra, J. F. and O. Talagrand, 1988: Short-range evolution of small perturbations in a barotropic model. *Tellus,* **40A**, 81-95.

[50] Le Dimet, F. X. and O. Talagrand, 1986: Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus*, **38A**,97-110.

[51] LeDimet F.X., H.E. Ngodock , B. Luong and J. Verron, 1997: Sensitivity analysis in variational data assimilation. *J Meteorol Soc Japan*, **75: (1B)**, 245-255.

[52] Le Dimet , F.X.and I Charpentier,1998: Methodes de second order en assimilation de donnees.In *Equations aux dérivées partielles et applications (articles dédiées à Jacques-Louis Lions)* , Gauthier-Villars, Paris 623-640.808pp.

[53] Lehoucq R.B., D.C. Sorensen and C. Yang, 1998: Arpack Users' Guide: Solution of Large-scale eigenvalue Problems with Implicitly Restarted Arnoldi Methods. Software, Environments, and Tools 6 ,SIAM, Philadelphia, 160pp.

[54] Lewis, R.M.,1997:A Nonlinear Programming perspective on sensitivity calculations for systems governed by state equations.ICASE Report No. 97-12,35pp.

[55] Liu, D. C and Jorge Nocedal, 1989: On the limited memory BFGS method for large scale minimization. *Math. Prog.*, **45**, 503-528.

[56] Moré, J.J., B. S. Garbow and K.E. Hillstrom, 1981:Testing Unconstrained Optimization Software. *ACM Transactions on Mathematical Software*, **7**, 17-41.

[57] Nash, S. G., 1982: Truncated-Newton methods. *Ph.D. Thesis,* Computer Science Department, Stanford University, CA.

[58] Nash, S. G., 1984: Newton-type minimization via the Lanczos method. *SIAM J. Numer. Anal.*, **21** (4), 770-788.

[59] Nash, S. G., 1984: Truncated-Newton methods for large-scale function minimization. *Applications of Nonlinear Programming to Optimization and Control,* H.E. Rauch (ed.), Pergamon Press, Oxford, 91-100.

[60] Nash, S. G., 1984: User's guide for TN/TNBC: Fortran routines for nonlinear optimization. *Tech. Rep. No.*, **397**, 17pp.

[61] Nash, S. G., 1984: Solving nonlinear programming problems using truncated-Newton techniques. *Numerical optimization,* P. T. Boggs, R. H. Byrd and R. B. Schnabel (ed.), SIAM, Philadelphia, 119-136.

[62] Nash, S. G., 1985: Preconditioning of truncated-Newton methods. *SIAM J. Sci. Stat. Comput.*, **6** (3), 599-616.

[63] Nash, S. G. and A. Sofer, 1989: Block truncated-Newton methods for parallel optimization. *Math. Prog.*, **45**, 529-546.

[64] Nash, S. G. and A. Sofer, 1989: A parallel line search for Newton type methods in computer science and statistics. *Proceeding 21-st Symposium on the Interface*, K. Berk and L. Malone (ed.), ASA, 134-137.

[65] Nash, S.G. and J. Nocedal,1991: A Numerical Study of the Limited Memory BFGS Method and the Truncated-Newton Method for Large Scale Optimization,*SIAM J. Optimization*,**Vol 1**, pp. 358-372.

[66] Nash, S. G. and A. Sofer, 1990: Assessing a search direction within a truncated-Newton method. *Operations Research Letters,* **9** (4), 219-221.

[67] Nash S.G.,2000: A Survey of Truncated-Newton Methods,*Journal of Computational and Applied Mathematics*, **124**, 45-59.

[68] Nash, S. G. and A. Sofer, 1996: Linear and Nonlinear Programming, McGraw-Hill (New York),ISBN 0-07-046065-5.

[69] Navon, I. M. and D. M. Legler, 1987: Conjugate gradient methods for large scale minimization in meteorology. *Mon. Wea. Rev.*, **115**, 1479-1502. conjugate gradient method

[70] Navon, I. M. , X. L. Zou, J. Derber and J. Sela, 1992: Variational data assimilation with an adiabatic version of the NMC Spectral Model.

Mon. Wea. Rev.,**122**, 1433-1446.

[71] Ngodock, H.E.,1996:Data Assimilation and sensitivity Analysis.Ph. D. Thesis, LMC-IMAG Laboratory, University Joseph Fourier, Grenoble, France, 213pp.

[72] Nocedal, J., 1980: Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, **35**, 773-782.

[73] Nocedal,J. and S.J. Wright,1999: Numerical Optimization.(Springer Series in Operations Research. Ed.: P. Glynn) Springer Verlag Series in Operations Research,656 pp., 85 illus.

[74] O'Leary, D. P., 1983: A discrete Newton algorithm for minimizing a function of many variables. *Math. Prog.*, **23**, 20-23.

[75] Park, S.K., and E. Kalnay, 1999: Application of the quasi-inverse method to storm-scale data assimilation. Abstracts, 5th SIAM Conference on Mathematical and Computational Issues in the Geosciences, March 24-27, 1999, San Antonio, Texas, SIAM, 104.

[76] Powell M.J.D., Toint P.L.,1979: Estimation of sparse Hessian matrices. SIAM J NUMER ANAL 16: (6) 1060-1074.

[77] Pu,Z.X. and E. Kalnay,1999: Targeting observations with the quasi-inverse linear and adjoint NCEP global models: Performance during FASTEX.*Quart. J. Roy. Meteor. Soc.*, **Vol. 125 No. 561**,pp 3329-3337

[78] Pu Z.X., E. Kalnay , J. Sela , et al.,1997: Sensitivity of forecast errors to initial conditions with a quasi-inverse linear method *Mon. Wea. Rev* ,**125: (10)**, 2479-2503.

[79] Rabier, F. and P. Courtier , 1992:Four dimensional assimilation in the presence of baroclinic instability.*Quart. J. Roy. Meterol. Soc.*, **Vol 118**, 649-672.

[80] Rabier, F., J-F. Mahfouf, M. Fisher, H. Jarvinen, A. Simmons, E. Andersson, F. Bouttier, P. Courtier, M. Hamrud, J. Haseler, A. Hollingsworth, L. Isaksen, E. Klinker, S. Saarinen, C.Temperton, J-N. Thepaut, P. Unden, and D. Vasiljevic,1997: Recent experimentation on 4D-Var and first results from a Simplified Kalman Filter. October 1997.ECMWF Technical Memorandum No. 240.

[81] Reuther, J. J.,1996: Aerodynamic Shape Optimization Using Control Theory.Ph.D. Dissertation, University of California, Davis, 1996.

[82] Santosa, Fadil and William W. Symes, 1988: Computation of the Hessian for least-squares solutions of inverse problems of reflection seismology. *Inverse Problems*, **4**, 211-233.

[83] Santosa, Fadil and William W. Symes, 1989: *An analysis of least squares velocity inversion.* Society of Exploration Geophysicists, Geophysical monograph #4, Tulsa.

[84] Schlick, T. and A. Fogelson, 1992: TNPACK–A truncated Newton minimization Package for large-scale problems: I. Algorithm and usage. *ACM Trans on Math. Soft.*, **18** (1), 46-70.

[85] Schlick, T. and A. Fogelson, 1992: TNPACK–A Truncated Newton minimization package for large-scale problems: II. Implementation examples. *ACM Trans Math. Soft.*, **18** (1), 71-111.

[86] Shanno, D. F. and K. H. Phua, 1980: Remark on algorithm 500 – a variable method subroutine for unconstrained nonlinear minimization. *ACM Trans. on Math. Soft.*, **6**, 618-622.

[87] Sleijpen G.L.G., H.A. Van der Vorst ,1996: A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J Matrix Anal A* **17: (2)**, 401-425 .

[88] Stewart III, G. W., 1967: A modification of Davidon's minimization method to accept difference approximations of derivatives. *Journal of the Association for Computing Machinery,* **14** (1), 72-83.

[89] Symes, William W., 1990:Velocity inversion: A case study in infinite-dimensional optimization. *Math. Programming*, **48**, 71-102.

[90] Symes, William W., 1991: A differential semblance algorithm for the inverse problem of reflection seismology. *Computers Math. Applic.*, **22**(4/5), 147-178. Math.,

[91] Symes, William W., 1993:A differential semblance algorithm for the inversion of Multioffset Seismic Reflection Data. *Jour. Geophys. Res.*, **98 B2**, 2061-2073.

[92] Talagrand, O. and P. Courtier, 1987: Variational assimilation of meteorological observations with the adjoint vorticity equation-Part 1. Theory. *Quart. J. Roy. Meteorol. Soc.*, **113**, 1311-1328.

[93] Thacker, W. C.,1989: The Role of Hessian Matrix in Fitting models to Measurements. *J. Geophys. Res.*, **94**, 6177-6196.

[94] Thomas, R. Cuthbert, Jr., 1987: *Optimization using personal computers.* John Wiley & Sons, New York, pp. 474.

[95] Veerse F.,1999: Variable-storage quasi-Newton operators as inverse forecast/analysis error covariance matrices in data assimilation. INRIA Technical Report No. 3685, Theme 4,28pp.

[96] Wang, Z., I. M. Navon, F. X. Le Dimet and X. Zou, 1992: The Second Order Adjoint Analysis: Theory and Application. *Meteorology and Atmospheric Physics*, **50**, 3-20.

[97] Wang, Z., I. M. Navon and X. Zou, 1993: The adjoint truncated Newton algorithm for large-scale unconstrained optimization. *Tech. Rep. FSU-SCRI-92-170,* Florida State University, Tallahassee, Florida, 44 pp.

[98] Wang, Z., 1993: Variational data assimilation with 2-D shallow water equations and 3-D FSU global spectral models. *Tech. Rep. FSU-SCRI-93T-149,* Florida State University, Tallahassee, Florida, 235 pp.

[99] Wang, Z., I.M.Navon, X. Zou and F.X. Le Dimet,1995: A Truncated -Newton Optimization Algorithm in Meteorology Applications with Analytic Hessian/vector Products, *Computational Optimization and Applications*,**volume 4** , 241-262.

[100] Wang, Z., Kelvin K. Droegemeier, L. White and I.M.Navon ,1997: Application of a New Adjoint Newton Algorithm to the 3-D ARPS Storm Scale Model Using Simulated Data, *Monthly Weather Review*,**Vol. 125, No 10**, 2460–2478.

[101] Wang, Z. , Kelvin K. Droegemeier and L. White, 1998 :The Adjoint Newton algorithm for large-scale unconstrained optimization in meteorology applications.*Computational Optimization and Applications*,**volume 10** , 283-320.

[102] Zou, X., I. M. Navon, F. X. Le Dimet, A. Nouailler and T. Schlick, 1990: A comparison of efficient large-scale minimization algorithms for optimal control applications in meteorology. *Tech. Rep. FSU-SCRI-90-167,* Florida State University, Tallahassee, Florida, 44pp.

[103] Zou, X., I. M. Navon, M. Berger, Paul K. H. Phua, T. Schlick and F. X. Le Dimet, 1993: Numerical experience with limited-Memory Quasi-Newton methods and Truncated Newton methods. *SIAM Jour. on Numerical Optimization*, **3**, 582-608.

[104] Zou, X., I. M. Navon and F. X. Le Dimet, 1992: Incomplete observations and control of gravity waves in variational data assimilation. *Tellus*, **44A**, 273-296.

[105] Zupanski, M. 1996 : A preconditioning algorithm for four-dimensional variational data assimilation. *Mon. Wea Rev.*, **124: (11)** ,2562-2573.

[106] Yang, W., I.M.Navon and P. Courtier,1996: A New Hessian Preconditioning Method Applied to Variational Data Assimilation Experiments using Adiabatic Version of NASA/GEOS-1 GCM . *Monthly Weather Review* **volume 124, No 5** 1000-1017.

Figures captions

Figure 1. The absolute (dashed line) and relative (solid line) differences between the Hessian/vector product computed with the SOA method and with the finite difference method at the first iteration (initial guess state). First 100 components are considered.

Figure 2. The evolution of the normalized cost function during the minimization using the SOA method (solid line) and the finite difference method (dashed line) to compute the Hessian/vector product

Figure 3. The evolution of the normalized gradient norm during the minimization using the SOA method (solid line) and the finite difference method (dashed line) to compute the Hessian/vector product

Table 1: First five largest and smallest computed Ritz values of the Hessian matrix and the corresponding relative residuals. The Hessian is evaluated at the initial guess point.

Largest values	Rel. residuals	Smallest values	Rel. residuals
5.29432E+02	1.74329E-06	3.19071E-02	3.04094E-03
4.87111E+02	2.18654E-06	6.02301E-02	2.85639E-03
4.35618E+02	1.79599E-06	7.77966E-02	1.44337E-03
3.86887E+02	2.03600E-06	7.83050E-02	1.99469E-03
3.81511E+02	1.80812E-06	9.16425E-02	1.75624E-03

Table 2: First five largest and smallest computed Ritz values of the Hessian matrix and the corresponding relative residuals. The Hessian is evaluated at the computed optimal point.

Largest values	Rel. residuals	Smallest values	Rel. residuals
5.12937E+02	1.63503E-06	1.46726E-02	5.87833E-03
4.73981E+02	1.54048E-06	1.71499E-02	1.60547E-02
4.19611E+02	1.73189E-06	4.56908E-02	3.07214E-03
3.88857E+02	1.70423E-06	7.22996E-02	1.89507E-03
3.78570E+02	1.89242E-06	8.28272E-02	3.00616E-03

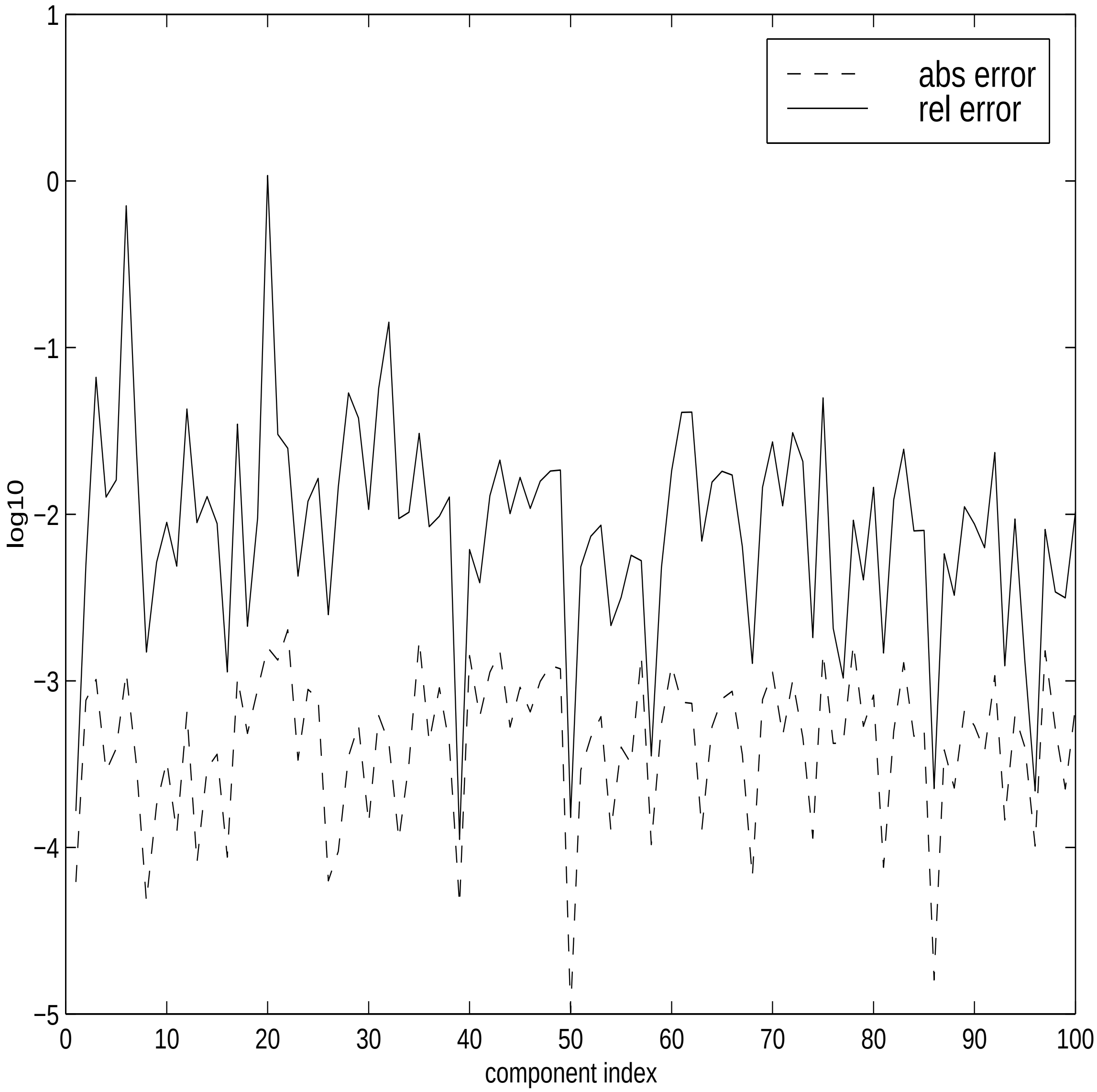

abs error
rel error
log10
component index
1
0
−1
−2
−3
−4
−5
0
10
20
30
40
50
60
70
80
90
100

Figure 1:

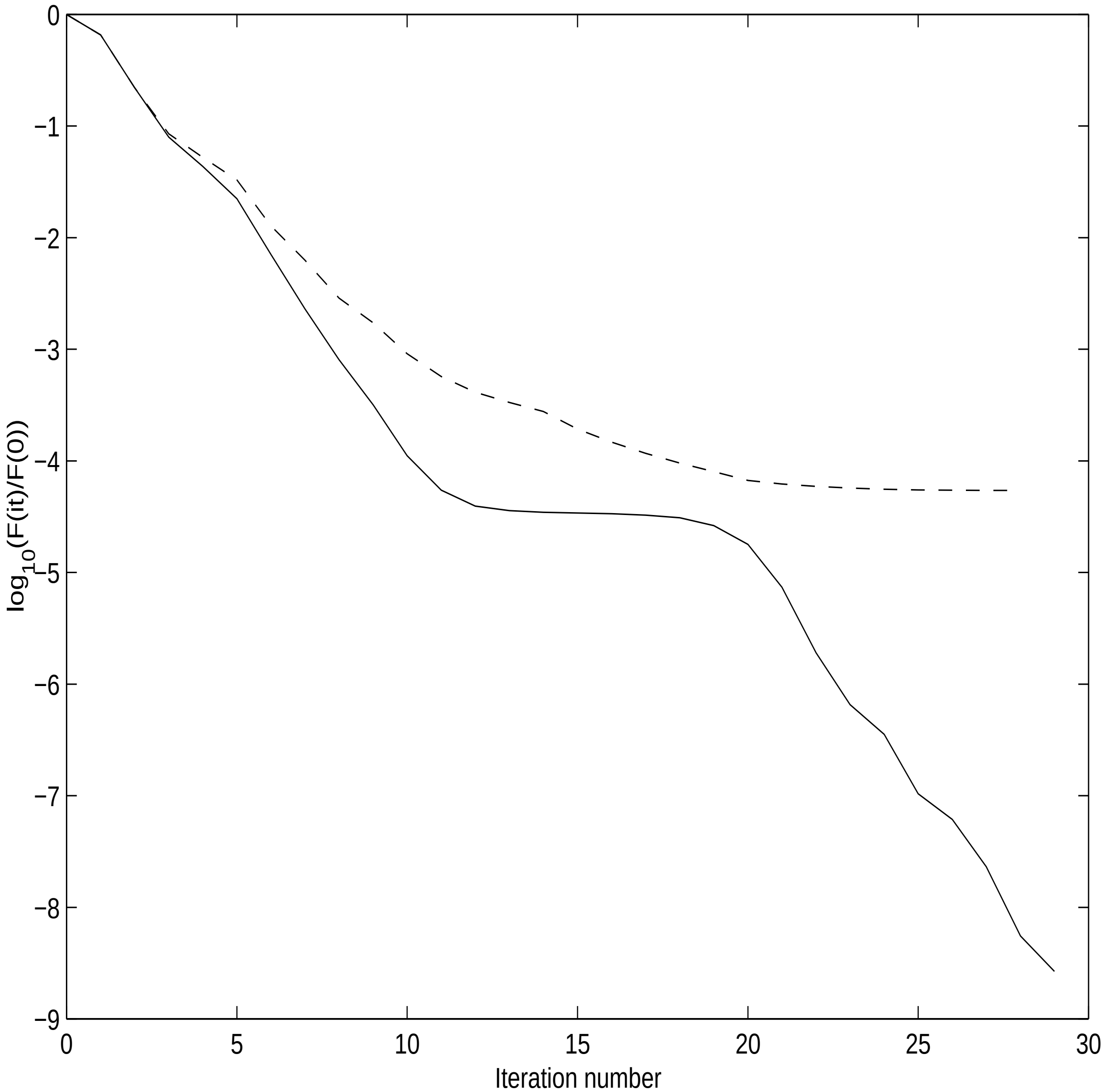
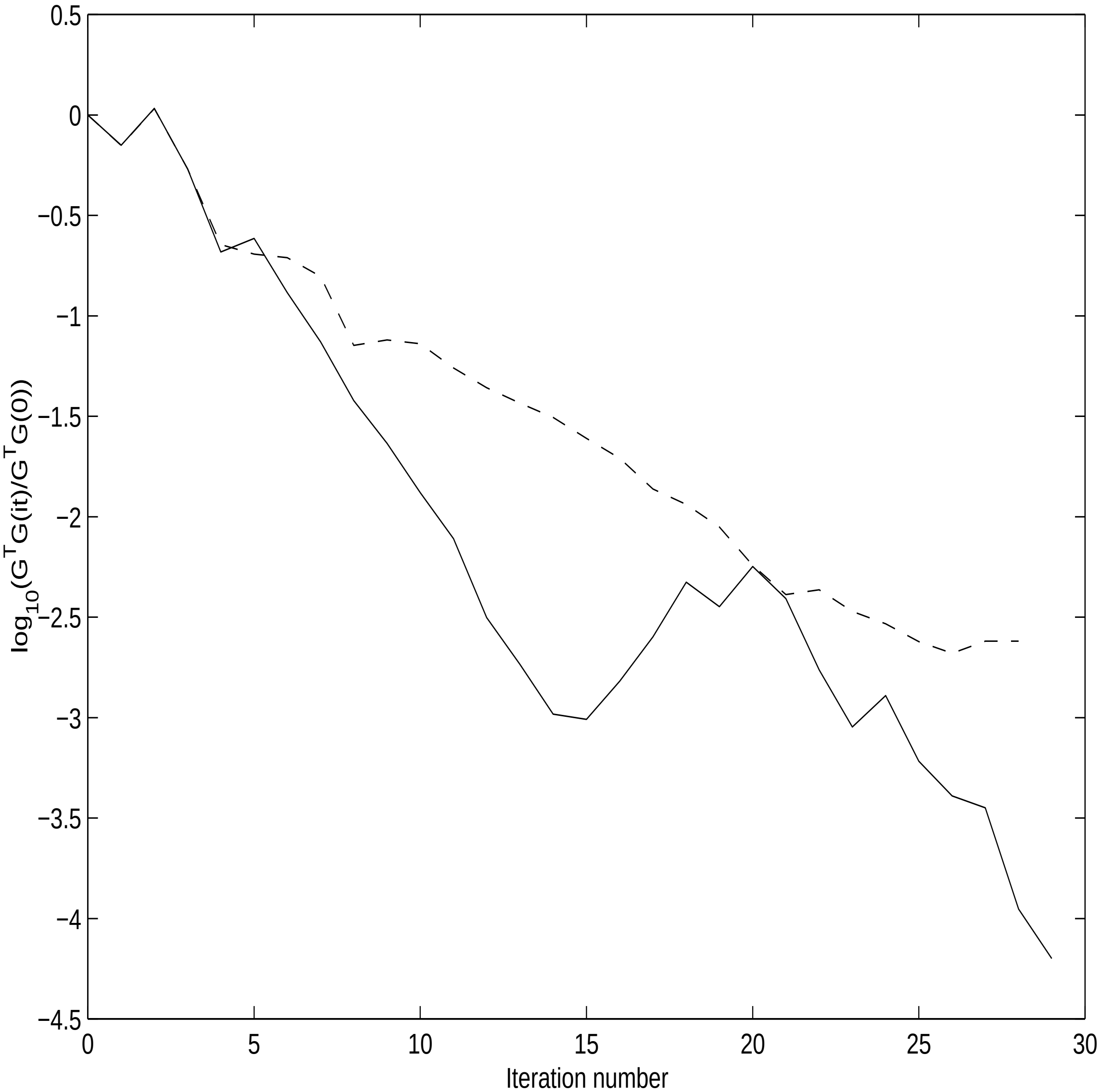


Figure 2:



0.5
0
−0.5
−1
−1.5
−2
−2.5
−3
−3.5
−4
−4.5
0
5
10
15
20
25
30
Iteration number
log10(G^T G(it)/G^T G(0))

Figure 3: