

NWPSAF-EC-UD-001

A One-Dimensional Variational Analysis Package

Frédéric Chevallier

ECMWF

This documentation was developed within the context of the EUMETSAT Satellite Application Facility on Numerical Weather Prediction (NWP SAF), under the Cooperation Agreement dated 25 November 1998, between EUMETSAT and the Met Office, UK, by one or more partners within the NWP SAF. The partners in the NWP SAF are the Met Office, ECMWF, KNMI and Météo France.

Copyright 2008, EUMETSAT, All Rights Reserved.

Change record			
Version	Date	Author / changed by	Remarks
1.0	11/1999	F. Chevallier	
1.1	01/2000	F. Chevallier	
1.2	05/2002	F. Chevallier	
2.0	11/2003	F. Chevallier	
2.1	03/2004	F. Chevallier	updated documentation from review (RID and B-test) suggestions
2.2	09/2004	A. McNally	updated documentation from DRI(re: V2.1 of software)
2.3	02/2008	A. McNally	updated documentation (re: V2.1 of software)

A one-dimensional variational analysis package

F. Chevallier *
Eumetsat/ECMWF NWP SAF

March 2004

1 Introduction

The present one-dimensional variational analysis (1D-Var) package is derived from a scheme that is used at the European Centre for Medium-Range Weather Forecasts (ECMWF) for data assimilation purposes. Examples of its application can be found in Marécal and Mahfouf (2000), Fillion and Mahfouf (2000), Chevallier and Mahfouf (2001) and Chevallier *et al.* (2002). It was re-written in 1999 to be delivered to Eumetsat in the framework of the Numerical Weather Prediction (NWP) Satellite Application Facility (SAF) and updated several times since.

Section 2 of the present documentation explains the principle of the scheme. The code is described in section 3. Advices to get started are given in section 4.

2 Description of the method

Given an array of background variables \mathbf{x}_b (i.e. the a-priori information), with an associated error covariance matrix \mathbf{B} , and given an array of observations \mathbf{y} , with an associated error covariance matrix \mathbf{R} , the present scheme determines statistically optimum values of the variables, \mathbf{x}_a . To do so, it minimises a cost-function which measures the fit to the background variables and to the observation:

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}_b) + \frac{1}{2}(H(\mathbf{x}) - \mathbf{y})^T \mathbf{R}^{-1}(H(\mathbf{x}) - \mathbf{y}) \quad (1)$$

where H is the observation operator (or forward model) that provides the model-equivalent of the observations.

The minimum is found by an iterative process, and not by any analytical expression. At each iteration, a descent direction is determined, using the value of the cost function gradient:

$$\nabla_{\mathbf{x}} J(\mathbf{x}) = \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}_b) + \mathbf{H}^T \mathbf{R}^{-1}(H(\mathbf{x}) - \mathbf{y}) \quad (2)$$

where \mathbf{H}^T is the adjoint operator of the Jacobian matrix \mathbf{H} of H (i.e. $\{\partial y_i / \partial x_j\}_{i,j}$).

For a better efficiency of the minimisation, \mathbf{x} is preconditioned with the eigenvectors of \mathbf{B} . The minimiser itself is a limited memory quasi-Newton method, the so-called M1QN3 software (version 2.0b dated December 1993) that has been developed at Institut National de Recherche en Informatique et en Automatique (INRIA). The reader is referred to Gilbert and Lemaréchal (1989) for its description.

*Current affiliation: LSCE, France, E-mail: frederic.chevallier@cea.fr

3 Description of the code

The package `1DVAR_2.1.tar` contains a series of Fortran 90 routines. They are controlled by a makefile: `1DVAR_2.1/Makefile`. In addition, two data-files are provided. One is a set of nine atmospheric profiles: `1DVAR_2.1/profiles.txt`. The other one is a coefficient file (`1DVAR_2.1/rt_coef_fmt.dat`), which is needed by the current observation operator.

3.1 The kernel

Compared to previous versions of this 1D-Var, the kernel of the package has been designed to fit any application with as few modifications as possible. This kernel includes the following routines:

- `1DVAR_2.1/src/module/onedvar_data.f90`
- `1DVAR_2.1/src/1dvar/onedvar.f90`
- `1DVAR_2.1/src/1dvar/simul.f90`

These routines should not be modified, except if the user needs to change the minimiser (in `simul.f90`), or the name of the observation operator function (in `onedvar.f90`).

`onedvar.f90` is the entry subroutine. `simul.f90` is needed by the minimiser to compute the value of the cost function and that of its gradient. `onedvar_data.f90` is a module that contains all the variables that `simul.f90` needs.

The arguments of `onedvar.f90` are:

inputs:

- `IPRINT` : integer. If `IPRINT` is not 0, then various prints are made on file unit `IPRINT`.
- `NSMAX` : integer. `NSMAX` is the maximum number of simulations that the minimiser is allowed to perform (user-defined).
- `SIMUL` : external function. An example is given in `1DVAR_2.1/src/1dvar/simul.f90`. The advantage of passing it as an argument is that several functions `simul` can be used in the same program, each one differing by a different observation operator (then the user has only to change the call to `obsop` in each function `simul`).
- `OBSOPER` : external function. `OBSOPER` is the direct observation operator. An example is given with `1DVAR_2.1/src/1dvar/obsop.f90`. It should be the same observation operator than the one called by external function `SIMUL`.
- `DIM_1D` : integer. `DIM_1D` is the dimension of `P1DIN`.
- `P1DIN` : one-dimensional real array (`DIM_1D`). `P1DIN` contains the initial values of the control variable array \mathbf{x} in equation (1). It should be set to the background value \mathbf{x}_b .
- `DIM_OBS` : integer. `DIM_OBS` is the dimension of `DIM_OBS`.
- `POBS` : one-dimensional real array (`DIM_OBS`). `POBS` is the variable array \mathbf{y} in equation (1).

- PEIG_VEC_1D : two-dimensional real array (DIM_1D×DIM_1D). PEIG_VEC_1D contains the eigenvectors of the matrix \mathbf{B} in equation (1). The vectors are along the columns (i.e. one vector per column). The units of \mathbf{B} have to be those of P1DIN.
- PEIG_VAL_1D : one-dimensional real array (DIM_1D). PEIG_VAL_1D contains the eigenvalues of the matrix \mathbf{B} in equation (1). The eigenvalues should be ordered in the same way than in PEIG_VEC_1D.
- PCOVAR_OBS_INV : two-dimensional real array (DIM_OBS×DIM_OBS). PCOVAR_OBS_INV contains the invert of the matrix \mathbf{R} in equation (1).

outputs:

- P1DOUT : one-dimensional real array (DIM_1D). P1DOUT contains the analysed values \mathbf{x}_a .
- XJS : value of cost function before the minimisation, $J(\mathbf{x}_b)$.
- XJE : value of cost function after the minimisation, $J(\mathbf{x}_a)$.

3.2 The observation operator

The subroutines that perform the forward computation $H(\mathbf{x})$ and the gradient computations $\mathbf{H}^T \mathbf{R}^{-1}(H(\mathbf{x}) - \mathbf{y})$ (optional) are:

- *1DVAR_2.1/src/modules/auxiliary.f90*
- *1DVAR_2.1/src/1dvar/obsop.f90*
- *1DVAR_2.1/src/rttovsrc/*.f90* (plus corresponding modules in *1DVAR_2.1/src/module/*)

1DVAR_2.1/src/1dvar/obsop.f90 is the interface between the forward model and the 1D-Var scheme. *1DVAR_2.1/src/1dvar/auxiliary.f90* is a module that is used by *obsop.f90*. It contains all the variables that are needed by the forward model but are not control variables (i.e. not in the P1DIN vector). These two subroutines have to be adapted for any application that differs from the demonstration one.

In the present demonstration set-up, the forward model is the RTTOV.v6 fast radiative transfer model (RTTOV: Eyre *et al.*, 1993; Saunders *et al.*, 1999) in a version that takes multi-layer cloudiness into account. A detailed description of that application can be found in Chevallier *et al.* (2002): *1DVAR_2.1/paper_gjrms.pdf*. The RTTOV routines are contained in directories *1DVAR_2.1/src/rttovsrc* and *1DVAR_2.1/src/module*.

The arguments of *obsop.f90* are:

inputs:

- DIM_1D : as above.
- DIM_OBS : as above.
- ZVAR_1D : input variables \mathbf{x} to the forward model H , organised as P1DIN.

- ZVAR_OBS_AD : input to the adjoint model. Consistently with Eq. (2), it contains $\mathbf{R}^{-1}(H(\mathbf{x}) - \mathbf{y})$ and is organised as POBS. Gradient computations are not performed if it is set to zero.

outputs:

- ZVAR_OBS : output variables, organised as POBS.
- ZVAR_1D_AD : output to the adjoint model, contains the gradient of the cost function with respect to the inputs, organised as P1DIN.

In the current setup, *obsop.f90* calls RTTOV to get the forward computation $H(\mathbf{x})$ and calls the adjoint of RTTOV to get the gradient computations $\mathbf{H}^T \mathbf{R}^{-1}(H(\mathbf{x}) - \mathbf{y})$.

Modifying the forward part of *obsop.f90* to fit some other application, should be trivial provided appropriate variables are stored in *auxiliary.f90*. Particular care should be given to the corresponding change in the gradient computations. Indeed the adjoint model (e.g. Errico 1997) is usually not available and finite differences have then to be used to compute the Jacobian matrix \mathbf{H} (\mathbf{H} should then be transposed and multiplied by ZVAR_OBS_AD).

3.3 Other subroutines

The main program of the demonstration version is:

- *main.f90*

main.f90 defines all the arrays that are needed by the subroutine *onedvar*. It has been designed consistently with Chevallier *et al.* (2002) and has to be replaced by the user's own program to fit his/her own application. The present version uses a simple interpolation routine (*1DVAR_2.1/src/util/rttov_intex.f90*).

As mentioned before, The 1D-Var uses \mathbf{B} 's eigenvector decomposition. This is done with the LAPACK library (*1DVAR_2.1/src/lapack*). It may be necessary to change the units of the control variables, in addition to the eigenvector-based preconditioning of \mathbf{x} which is performed in the code. Indeed, the eigenvector decomposition may not be accurate enough for the smallest values of \mathbf{B} .

The inversion of \mathbf{R} is done with *1DVAR_2.1/src/util/matinv.f90*.

Finally a set of subroutines contains the minimiser M1QN3 (INRIA copyright):

- *1DVAR_2.1/src/m1qn3/*.F*

The M1QN3 copyright file is also in *1DVAR_2.1/src/m1qn3/*.

Users outside the SAF partners shall return it to INRIA.

3.4 Data-flow

After the initialisations, *main* calls *onedvar*. *onedvar* calls *obsop*, *simul* and *m1qn3*. *simul* calls *obsop*. *obsop* calls *rttovcld* and *rttovcldad*. *simul* is also called from the *m1qn3* package.

4 To install and start the 1DVAR package

Here are the basic instructions to start the code in its test version:

- > *gzip -df 1DVAR_2.1.tar* : uncompress tar file
- > *tar -xf 1DVAR_2.1.tar* : creates a tree under directory *./1DVAR_2.1/*
- > *cd 1DVAR_2.1*
- > *make clean* : deletes previous objects.
- > *make* : builds objects.
- > *main* : runs 1DVAR.

As mentioned above, the program uses two input files in the current configuration: *profiles.txt* and *rt_coef_fmt.dat*.

It generates two output files:

- Since *IPRINT* is not zero, details of the minimisation are saved in file unit *IPRINT*, which is connected to *1dvar.log*. This file may be compared with the reference files for each one of the nine profiles: *ref/1dvar.log_prof**. Typically the results are physically reasonable if the figures are reproduced to four decimal places. These files have been generated with LINUX compiler ifort.
- The background, the observed and the analysed profiles are also saved in file *1dvar.out* for quick visualisation with the gnuplot software. If the latter is available, type 'gnuplot trace'.

The Makefile has been written for LINUX compiler ifort. Some modifications may be needed for other ones, in particular for the compilation in double precision (-r8 option), necessary for M1QN3 as it is set here (M1QN3 uses implicit double precision).

Acknowledgements

The present 1D-Var scheme is a new version of a scheme that was initially provided by Jean-François Mahfouf. Marco Matricardi, Jean-Noël Thépaut (ECMWF), Reinhold Hess (DWD), Godelieve Deblonde (MSC) and Anke Thoss (SMHI) made valuable comments to improve the presentation of the code. Thanks are also due to the Institut National de Recherche en Informatique et en Automatique, Rocquencourt, France, for providing the M1QN3 minimisation code. Specific rules apply for its use (see M1QN3 documentation).

References

Chevallier, F., and J.-F. Mahfouf, 2001: Evaluation of the Jacobians of infrared radiation models for variational data assimilation. *J. Appl. Meteor*, **40**, 1445-1461.

- Chevallier, F., P. Bauer, J.-F. Mahfouf and J.-J. Morcrette, 2002: Variational retrieval of cloud profile from ATOVS observations. *Q. J. R. Meteor. Soc.*, **128**, 2511-2526.
- Eyre, J. R., 1991: A fast radiative transfer model for satellite sounding systems. *ECMWF Technical Memorandum No. 176*.
- Errico, R. M., 1997: What is an adjoint model? *Bull. Amer. Meteor. Soc.*, **78**, 2577-2591.
- Fillion, L., and J.-F. Mahfouf, 2000: Coupling of moist-convective and stratiform precipitation processes for variational data assimilation. *Mon. Wea. Rev.*, **128**, 109-124.
- Gilbert, J. C., and C. Lemaréchal, 1989: Some numerical experiments with variable-storage quasi-Newton algorithms. *Mathematical Programming*, **45**, 407-435.
- Marécal, V., and J.-F. Mahfouf, 2000: Variational retrieval of temperature and humidity profiles from TRMM precipitation data. *Mon. Wea. Rev.*, **128**, 3853-3866.
- Saunders, R., M. Matricardi, and P. Brunel, 1999: An improved fast radiative transfer model for assimilation of satellite radiance observations. *Quart. J. Roy. Meteor. Soc.*, **125:556**, 1407-1425.