

ESA - ESRIN

Data User Programme

Project: DAMS2P (December 1996 – February 1998)

Final Report, April 1998

**DEVELOPMENT OF GLOBAL AEROSOL MAPPING
FROM SATELLITES LEVEL-2 PRODUCTS
(DAMS2P)**

Cartography and 4D Variational Data Assimilation

Key persons:

M. DE MAZIERE (co-ordinator)

D. FONTEYN

G. FRANSENS

D. FUSSEN

BIRA-IASB
Belgian Institute for Space Aeronomy, Ringlaan 3, B-1180 Brussels,
BELGIUM
Tel: +32 2 373 0401, Fax: +32 2 374 8423

TABLE OF CONTENTS

ABSTRACT	3
EXECUTIVE SUMMARY	4
1 INTRODUCTION	4
1.1 PURPOSE, SCOPE AND GOALS OF THE PROJECT	5
1.1.1 PROJECT DELIVERABLES	5
1.2 POTENTIAL USE OF THE PROJECT RESULTS	6
1.3 PROBLEM DESCRIPTION	6
1.4 TECHNICAL APPROACH	7
2 WORK PROGRAMME AND SUMMARY OF ACHIEVEMENTS	8
2.1 WORK PACKAGES: SUMMARY OF ACTIVITIES AND RESULTS	8
2.2 PUBLICATIONS AND PRESENTATIONS	10
2.3 USER CONTACTS	10
3 CARTOGRAPHY ALGORITHM	12
3.1 STRUCTURE OF THE SOFTWARE	12
3.2 DESCRIPTION OF PARAMETERS	14
3.3 LIMITATIONS OF THE INTERPOLATION ALGORITHM	17
3.4 PSEUDO CODE DESCRIPTION OF THE SOFTWARE	18
3.5 ALGORITHM VALIDATION	20
3.5.1 INTERPOLATION OF A KNOWN FUNCTION	20
3.5.2 INTERPOLATION ERROR CAUSED BY UNCERTAINTIES ON THE DATA	22
3.5.3 SELF CONSISTENCY OF THE INTERPOLATION	23
3.6 EXAMPLES OF AEROSOL MAPS FROM CARTOGRAPHY	24
3.6.1 SAGE II	24
3.6.1.1 Altitude sections for January 1992 (post Pinatubo period)	24
3.6.1.2 Altitude sections for January 1989 (pre Pinatubo period)	24
3.6.1.3 Stratospheric variability in 1992 at 26.5 km altitude	24
3.6.1.4 Total column optical thickness during the period August 15 – September 24, 1993	25
3.6.2 ORA	25
3.6.2.1 Total column optical thickness during the period September 1992 – April 1993	25
3.6.3 GOME	25
3.6.3.1 Total column optical thickness	26
4 ASSIMILATION OF STRATOSPHERIC AEROSOLS USING REMOTE SENSING OPTICAL DATA.	27
4.1 INTRODUCTION TO ADJOINT MODELS	27

4.1.1	ADJOINT MODELS AND THE ASSIMILATION CONCEPT	27
4.1.2	A DISCRETISED APPLICATION: 4D-VARIATIONAL DATA ASSIMILATION ON ATMOSPHERIC COMPOSITION	29
4.1.3	DEVELOPMENT OF THE ADJOINT MODEL CORE	31
4.2	INTRODUCTION TO 4D-VAR AEROSOL ASSIMILATION.	33
4.2.1	MONO-WAVELENGTH APPROACH.	34
4.2.2	SPATIAL AND TEMPORAL COVERAGE OF REMOTE SENSING DATA.	34
4.2.3	DESCRIPTION OF THE 4D-VAR ASSIMILATION MODEL.	35
4.2.3.1	Model description.	35
4.2.3.2	Adjoint model.	35
4.2.3.3	Optimisation algorithm.	35
4.2.4	THE FIRST GUESS OF THE MODEL INITIAL CONDITIONS.	35
4.2.5	AEROSOL ASSIMILATION FLOWCHART AND CODE DESCRIPTION	36
4.2.6	OUTPUT AND FILE FORMATS	38
4.2.7	IDL ASSIMILATION VISUALISATION AND ANALYSIS TOOL.	38
4.3	VALIDATION OF 4D-VAR ASSIMILATION FRAMEWORK.	39
4.4	DEFAULT SETTINGS FOR ASSIMILATION ANALYSIS.	39
4.5	DOMAIN OF VALIDITY OF THE ASSIMILATION ANALYSIS.	39
4.5.1	SPATIAL VALIDITY OF THE ASSIMILATION ANALYSIS.	39
4.5.2	TEMPORAL VALIDITY OF THE ASSIMILATION ANALYSIS.	40
4.6	RESULTS.	40
4.7	FUTURE IMPROVEMENTS.	41
4.8	CONCLUSIONS.	41
 5 MIGRATION TOWARD FUTURE ENVISAT DATA		 50
 6 CONCLUSIONS		 51
 7 BIBLIOGRAPHY		 52
 APPENDIX A. LIST OF ACRONYMS		 53
 APPENDIX B. CARTOGRAPHY ALGORITHM MATHEMATICAL DESCRIPTION		 54
B.1	THE INTERPOLATION PROBLEM	54
B.2	DOMAIN DECOMPOSITION	55
B.3	THE MULTIPOINT TAYLOR SERIES REPRESENTATION	55
B.4	ESTIMATION OF SPATIAL DERIVATIVES	58
B.5	FINAL SPATIAL FILTERING	60
B.6	FILE FORMAT DEFINITIONS	61
B.6.1	FORMAT 'PRF'	61
B.6.2	FORMAT 'ACT'	64
B.6.3	FORMATS 'SAM' AND 'INT'	64

Abstract

The current report discusses the development of tools for the production of global aerosol maps starting from satellite level-2 aerosol data. This work was the subject of the project DAMS2P, Development of Global Aerosol Mapping from Satellites level-2 Products, in the frame of the ESA Data User Programme (DUP). Two approaches have been pursued: a low-level cartography approach, which essentially is based on interpolation, and a 4D variational data assimilation approach, which produces a map that is internally consistent with the atmospheric transport model upon which the assimilation algorithm is based. The algorithms characteristics and software structure are described, together with the user parameters, limits of applicability and an evaluation of the reliability of the results (algorithm validation). Examples of aerosol maps produced by both techniques are presented.

The techniques developed here are promising for a variety of satellite data users. They will be refined and extended to additional applications and future satellite missions like ENVISAT, in the successor project AMASDU (Aerosol Mapping Algorithms for Satellite Data Users, 1998-).

Executive Summary

The wealth of observational data that is becoming available from satellite experiments is growing steadily, following the increasing complexity of scientific knowledge and associated research tools. Potential users of satellite data are thus confronted with a problem of accessibility and visibility of these data. The ESA DUP Programme addresses that problem: its goal is the enhancement of the exploitation of the satellite data, through the development of tools that come up to the requirements of potential users.

In this context, the DAMS2P (Development of Global Aerosol Mapping from Satellites level-2 Products) objectives are the development of mapping tools for satellite level-2 data; at present aerosol data are considered, but the results will be extended in the future to additional atmospheric data products. Examples of potential users of the resulting global aerosol maps are the remote sensing community, satellite validation teams, study groups for the assessment of the atmospheric aerosol load and evolution, etc.

Two different approaches have been followed for the production of global aerosol maps: a low-level cartography approach, and a 4D variational data assimilation method. The former approach is based essentially on interpolation, the latter one on a numerical atmospheric model that propagates the aerosol particles in a dynamically consistent way.

Regarding the cartography approach, a new interpolation algorithm has been developed, based on Multipoint Taylor expansion in 2D. It has been shown that this interpolation algorithm is better adapted to the specific problems of satellite data (sparse data set, random sampling grid, data uncertainties, etc.) than commonly used algorithms. A local thin plate spline fit allows the estimation of spatial derivatives in the interpolation formula, taking into account that the data may have important uncertainties. The interpolation is followed by a smoothing operation (low-pass spatial frequency filtering) for the removal of unphysical fast variations in the resulting maps. The amount of smoothing is one parameter out of a number of user parameters that allow to adapt the mapping results to specific user needs and/or data characteristics. Actually, the data considered are aerosol optical extinction coefficients at one wavelength. The mapping is done at a certain altitude. Else, total column extinction maps can be produced as well.

Regarding the assimilation approach, the algorithm has been built on an existing stratospheric transport model of the stratosphere. Aerosol particles are considered as inert tracers. They are propagated through the space state, and the initial global aerosol state is derived iteratively according to the criterium that it is the one that best fits with the observations, in each space-time point encountered within the time window considered. The model dynamics are based on ECMWF wind analyses. At best, the time resolution is of order 30', the spatial resolution is of order $1.8^\circ \times 1.8^\circ$; 5 stratospheric levels are considered. Currently, the aerosol particle density is assumed to be proportional to the extinction coefficient. It turns out that the initial guess of the initial aerosol state is critical for the convergence characteristics of the method. At present this is solved by using a back-trajectory calculation starting from the observations which gives the most satisfactory results. The results show that the assimilation analysis is capable of reproducing to a high degree the observed optical extinction data. In addition, preliminary results show that non-aerosol events can be detected by this assimilation technique.

The current developments are based mainly on SAGE-II aerosol data; some efforts have been devoted to ORA/Eureca and GOME/ERS-2 data. A number of example aerosol maps produced by both methods are discussed.

Both mapping algorithms have been validated through various tests. In the near future, mutual intercomparisons will give an even better estimate of the ultimate reliability of both methods, in various circumstances. The tools will be refined and expanded to additional applications (multi-wavelength approach, additional species,...) and future satellite missions like ENVISAT, in the successor project AMASDU (Aerosol Mapping Algorithms for Satellite Data Users, 1998-). The development done here is such that the extension to other satellite data, e.g., ENVISAT, should be straightforward.

1 Introduction

1.1 Purpose, scope and goals of the project

It is being recognised widely that the latest scientific space experiments have been very successful in the area of Earth observations. The databases related to the Earth environment have expanded rapidly during the last years, in volume as well as in diversity. Satellite platforms such as UARS and the planned ENVISAT mission contain several instruments, based on different techniques, which operate simultaneously to obtain quasi-identical, complementary and/or correlated information.

In order to let users cope with this vast amount of data, it is realised:

- (i) That a serious effort is required to make these data more easily accessible and more comprehensible (e.g. , through visualisation) for a wide user-community, and
- (ii) that added-value data can be offered to the users if data from various instruments/platforms are combined into coherent sets at various levels of interpretation, suitable for use by modellers, climatologists, policy makers, etc.

Such ideas are underlying the establishment of the ESA Data User Programme (DUP), into which this project is circumscribed.

It has become ever more clear over the last years that the aerosol load of the atmosphere, in the troposphere as well as in the stratosphere, plays a fundamental role in the photo-chemical and radiative equilibrium of the Earth atmosphere. The eruption of Mt. Pinatubo and the enhanced capabilities for atmospheric observations and modelling have triggered our understanding of the impact of aerosols on the Earth climate and environment. Aerosols have both direct and indirect impacts on these. They have a direct effect on the radiative budget of the Earth by increasing the total optical extinction. An example of an indirect effect is their interaction with ozone, through heterogeneous chemistry processes on aerosol and PSC particles. The concentrations of chemical reservoir species like HNO_3 , N_2O_5 and ClONO_2 are good indicators of such heterogeneous processes. The effectiveness of these atmospheric processes is largely influenced by the aerosol properties such as particle density, size distribution, refractive index and composition.

The current project aims at defining and developing cartography tools for integrating satellite level-2 aerosol data from several 'sources', with various characteristics, and to interpolate them towards a common grid. In parallel, aerosol data assimilation in an atmospheric transport model is developed. The results are quasi-global maps of aerosol data, with different temporal resolutions. The assimilation tool is restricted to stratospheric aerosol because of the transport model which it is based upon.

Both developments are based on aerosol data from past and present missions that are available already now, as e.g., SAGE-II (ERBS), ORA (EURECA) and UARS. The application to GOME (ERS-2) and POLDER (ADEOS) data is included, to the extent possible (availability and reliability of data, appropriateness of the tools, etc.) Currently, long-term (17 years) global stratospheric aerosol data sets are limited to the SAM II, SAGE-I and SAGE-II experiments. More recently, UARS is providing an interesting multiple aerosol data set (spanning 6 years by now). ENVISAT will follow on after 1999. No comparable tropospheric data are available yet.

The end goal of this project is to dispose of tools that enable us to deliver global (assimilated) aerosol maps based on available satellite level-2 aerosol data. These tools are being prepared with sufficient generality in order to map future ENVISAT aerosol products; the extension of the tools to enable also the mapping of other data such as water vapour and chemical reservoir species, is projected.

1.1.1 Project deliverables

The deliverables of this project are

- the description of both the cartography and assimilation algorithms for mapping
- examples of global maps of aerosol data, resulting from cartography on the one hand and from 4D variational data assimilation on the other hand.
- a description of user parameters and interfaces

1.2 Potential use of the project results

- Many remote sensing experiments are hindered to some extent by the presence of aerosols. These require a better characterisation of the aerosol distribution and especially of their optical properties.
- The evolution of the global aerosol load, the occurrence of PSCs and heterogeneous processing should be monitored. It is affected by anthropogenic activities as well as natural events, e.g., biomass burning, aircraft-fleet expansion, volcanic eruptions and climate (temperature) changes.
- The aerosol climatological models currently in use (e.g. in MODTRAN) could be improved by the full exploitation of current long-term and future observational data.

Therefore, the output of this project will be beneficial for a widespread community, as summarised hereafter.

a) In the short term

- (i) for Earth's surface remote sensing applications: for correcting the retrieved information in optical channels for aerosols;
examples: JRC (Joint Research Centre), ISPRA, European Environment Agency.
- (ii) for the climate and weather modelling community: through the chemical and radiative properties of aerosols;
examples: IPCC Assessment Groups.
- (iii) for UV-monitoring, modelling and prediction: because these need the aerosol optical properties;
example: IPCC Assessment Groups, European Environment Agency.
- (iv) for validation purposes of satellite measurements of atmospheric constituents; the use is currently limited to tracers but the future project AMASDU will expand this to chemical species as well.
example: GOME and ENVISAT validation teams.

b) In the long term

- (i) for the assessment of the long term evolution of aerosol load and PSC frequency of occurrence: their impact on long term trends of ozone and the stratospheric composition;
- (ii) for the assessment of influence of volcanic eruptions under changing stratospheric composition;
- (iii) for the assessment of impact of a commercial subsonic/supersonic aircraft fleet;
examples: WMO, UNEP assessment groups, EC DGXII .
- (iv) for policy-makers and for establishing international research policies (e.g. EC DGXII);
example: for evaluation of Montreal Protocol.

1.3 Problem description

Global mapping of aerosol data based on satellite level-2 data is not evident for the following reasons:
data sampling:
the sampling of the level-2 data is irregular; this sampling must be transformed to a representation on a common regular grid on the globe
data coverage:

the raw data have a limited coverage of the globe: it depends on the length of the time window of sampling to which extent the coverage gets complete. For example, for the SAGE-II data, global coverage is obtained after about one month sampling.

depending on the chosen time period, the coverage is more or less complete as to latitude

depending on the chosen time period, the data set may show important gaps where data are completely missing or where data are unreliable

data uncertainties

the data are contaminated by errors: the latter may be indicated as an error bar on the data; sometimes data points are clearly aberrant. In addition regarding aerosols, data are contaminated by the presence of clouds and PSCs. In case of high aerosol loading of the atmosphere (volcanic perturbation), SAGE-II data are lacking because the atmosphere becomes opaque for the instrument (measurements in the solar occultation mode).

data of limited information: out of the various aerosol properties (particle density, composition, optical characteristics, size distribution,...), mostly one parameter only is available. Therefore one must make assumptions (an aerosol model) on the purpose of deriving additional properties if needed. For example in case of SAGE-II, the observations concern the optical extinction β (1/km) at the wavelengths of 525 nm and 1020 nm whereas the assimilation approach needs particle number densities: a simple aerosol model will be used to make the conversion. In this project, no attempt is made yet to derive the aerosol properties from data at different wavelengths, possibly from different satellite experiments.

1.4 Technical approach

Two different approaches have been pursued: (i) a low-level cartography approach (WP 1100), and (ii) a 4-D variational data assimilation approach (WP 1200).

(I) low-level cartography:

This approach aims at the development of interpolation tools that address the problems cited in the previous section. The result will be a global map with a limited time resolution which is about the time the satellite experiment needs to obtain global coverage. Gaps are filled unless they are too large, because then the confidence level of the result decreases. As said before, the maps are single wavelength; merging of data from different experiments has not been attempted yet.

(ii) 4D variational data assimilation:

This second approach aims at the production of global maps that are consistent with the atmospheric behaviour following a numerical model of the atmosphere. In the case of aerosol in normal conditions, it is a good approximation to consider them as inert tracers and therefore to restrict the atmosphere model to a pure transport model. In perturbed conditions (PSC formation, volcanic eruptions,...), the chemistry and microphysics of the aerosol should be considered also; such a situation is not yet addressed in the current project.

Therefore, the aerosol assimilation maps have internal dynamical consistency and are of a higher temporal resolution than the cartography results.

As said before, the derivation of the aerosol particle density, which is the physical quantity that is transported in the assimilation model, is based on a most simple aerosol model; no attempt is made yet to derive the aerosol properties from data at different wavelengths, and/or from different satellite experiments.

2 Work programme and summary of achievements

2.1 Work packages: summary of activities and results

WP 1000 Project Management

M. De Mazière

The Kick-Off meeting was held in ESRIN, on March 3, 1997; the corresponding minutes were delivered to ESRIN on March 6, 1997.

Regular interim reports (with approximately 6 weeks intervals) have been submitted to ESRIN by e-mail.

A Mid-Term meeting was held at BIRA-IASB, on September 30, 1997; a corresponding progress report was delivered to ESRIN on November 21, 1997.

The final meeting was held in ESRIN, on March 30, 1998, including delivery of a draft version of the final report. The minutes of it were delivered to ESRIN on April 3, 1998. It was agreed to deliver the final report by the end of April, 1998.

The team has convened regularly for discussions about encountered problems and intermediate results, and for exchanging ideas.

WP 1110/WP 1210 Experimental data compilation

M. De Mazière / P. C. Simon

A table comprising an overview of existing and near-future satellite level-2 aerosol products was assembled: it was included in the Annex to the Kick-Off meeting minutes.

The majority of the development and tests performed within WP 1100 and WP1200 have been using the SAGE-II database that is being maintained at the institute. The main reasons for this choice are that

- (i) these data are easily accessible,
- (ii) they have been studied earlier at the institute regarding their reliability, variability, etc., and
- (iii) they cover a long-time period (1984 → now) including the pre- and post-Pinatubo periods, and they have a near global spatial coverage. The data used in the cartography algorithm development up to now are the SAGE-II vertical profiles of aerosol extinction ratio (β), at wavelengths 0.385, 0.453, 0.525 and 1.020 μm .

WP 1100 Development of cartography

G. Franssens / D. Fussen

A review of some of the available 2D interpolation algorithms was carried out [Davis, 1975; Dierckx, 1995]. This evaluation was not an exhaustive or thorough comparison, but rather a limited assessment in view of the requirements posed by a cartography of aerosol data.

An evaluation of the following methods was performed:

- (i) Delaunay triangulation + linear interpolation
- (ii) Kriging interpolation
- (iii) Cubic splines with tension

- (iv) Spherical harmonics expansion
- (v) A new method based on Multipoint Taylor series

For details about the results of this evaluation we refer to the mid-term report.

In addition to the existing methods (i-iv), a new method (v) was developed based on the new concept of Multipoint Taylor (MT) series. The design of a new method was motivated by

- (i) unsatisfactory results
- (ii) limitations of applicability
- (iii) lack of robustness
- (iv) lack of controllability

of the four existing methods. The majority of the effort devoted to this work package went into the development, fine tuning and testing of this new algorithm.

The new method was designed to perform interpolation of randomly scattered data directly on the sphere and not after projection onto a plane. Furthermore, the new method solves the more general Hermite interpolation problem (using functional values as well as spatial derivatives) instead of the usual Lagrange problem (only functional values). Another important advantage is that its computational cost is linear in time with the number of sample points. Finally the resulting interpolant has superior analyticity properties compared to most existing interpolation methods.

The new algorithm has been applied to a number of typical SAGE II data sets, including sparse data, uniform background mode and high volcanic aerosol mode. It has also been extensively tested as to its reconstruction capabilities of *a priori* known real-analytic functions sampled over regular and irregular, SAGE type, grids. Validation tests and an error bound analysis of the algorithm have been carried out. A typical interpolation example uses a SAGE-II coverage of one month, which is equivalent to about 900 measurement points, at a chosen altitude. The algorithm interpolates towards a fine grid of 360*181 points. Average computing time for producing a single 2D map is typically of the order of 30 seconds on a shared HP SMP K-series UNIX server.

WP 1200 Assimilation of stratospheric aerosol in transport models

D. Fonteyn / J-F. Muller / Q. Errera

A review of some available assimilation algorithms was carried out. From this review it was clear that the most important advance in assimilation techniques has been made in the domain of numerical weather prediction and that assimilation techniques are just beginning to be used in the field of stratospheric composition.

One of the most promising assimilation techniques is the four-dimensional variational assimilation technique (4D-Var). This technique consists in finding model initial conditions which minimise the misfit between model values and observations for a given period. The algorithm requires a model for the description of the evolution of the quantities to be assimilated (or analysed), the gradient of the misfit with respect to the model initial conditions (adjoint model) and a minimisation algorithm.

We developed an aerosol assimilation concept based on the 4D-Var assimilation technique. In this concept we already prepared the inclusion of a more detailed description of aerosols and the inclusion of chemical species in the assimilation process. The concept relies on modular model design and extensive use of automatic differentiation.

The implementation of the 4D-Var assimilation of stratospheric aerosols using a mono-wavelength approach is discussed and the advantages and limitations are illustrated by the results obtained by the SAGE-II analyses. The results show that 4D-Var is able to reproduce the observations (in this case SAGE-II).

2.2 Publications and presentations

- The 1D development of the new interpolation method based on Multipoint Taylor series has been submitted to the journal “Advances in Computational Mathematics”, as G. Franssens, “A new non-polynomial univariate interpolation formula of Hermite type”.
- The new interpolation method has been presented by G. Franssens, as “Real-analytic interpolation based on Multipoint Taylor series”, at the Joint Conference of the AMS, the SAMS and the LMS, Special Session on Numerical Analysis and Approximation, Pretoria, SA, 25-28 June, 1997.
- A number of mapping examples, obtained with the spherical interpolation algorithm in combination with transport assimilation, has been presented by G. Franssens, as G. Franssens, D. Fonteyn, M. De Mazière and D. Fussen, “Cartography of the SAGE-II aerosol products using assimilation and a new spherical interpolation method”, at the Workshop on Inverse Methods in Global Biogeochemical Cycles, Heraklion, Crete, Greece, 16-20 March, 1998. The proceedings will be published as a peer-reviewed AGU Monograph.
- The 2D spherical interpolation algorithm will be presented by G. Franssens, as “A new solution to the Hermite-Birkhoff interpolation problem on the sphere, with applications in geosciences”, at the South African Symposium on Numerical Mathematics 98, Cape Town, SA, 15-17 April, 1998.

It is expected that the application of the new cartography algorithm in combination with a full 4D variational data assimilation will lead to multiple additional publications in the near future.

2.3 User contacts

The following persons expressed their interest in either the methods developed in the current project, or in possible aerosol level-3 products to be delivered at a later stage.

1. Daniel Heyndrickx,
Magnetosphere Research Group,
Belgian Institute for Space Aeronomy,
Brussels, Belgium.
Interest: interpolation.
2. Jean-Christoffer Lambert,
GOME Group,
Belgian Institute for Space Aeronomy,
Brussels, Belgium.
Interest: interpolation, assimilation.
3. Olivier Arino,
MERIS Group,
ESA/ESRIN,
Frascati, Italy.
Interest: aerosol level-3 products.
4. Ricardo Todling,
Data Assimilation Office,
NASA/GSFC,
Greenbelt, MD, USA.
Interest: interpolation.
5. Lori Bruhwiler,
Environmental Research Laboratories,

NOAA/CMDL,
Boulder, CO, USA
Interest: aerosol level-3 products.

6. Guus Velders,
Air Research Laboratory,
National Institute of Public Health and the Environment,
Bilthoven, The Netherlands.
Interest: aerosol level-3 products.
7. Bob Swap,
Global Environmental Change Program,
University of Virginia,
Charlottesville, VA, USA.
Interest: interpolation, assimilation, aerosol level-3 products.
8. Mahesh Phadnis,
CGRER,
University of Iowa,
Iowa City, IA, USA.
Interest: interpolation, assimilation.
9. Shoichi Taguchi,
Environmental Assessment Division,
National Institute for Resources and the Environment,
Ibaraki, Japan.
Interest: interpolation, assimilation

3 Cartography algorithm

3.1 Structure of the software

The software tools developed within WP1100 and interfacing with WP1200 have been structured in a modular way. These tools form the interface between the various tabulated level-2 data and the final visualisation of the produced maps.

The cartography process naturally splits up into four independent processing layers (Fig. 3.1). Data flows from the top, the aerosol data sources, to the bottom, the graphical files storing the visualisation. The four processing layers contain the following:

- (i) data selection and file format converting tools,
- (ii) data processors
- (iii) interpolation routines
- (iv) visualisation and graphical file formatting tools

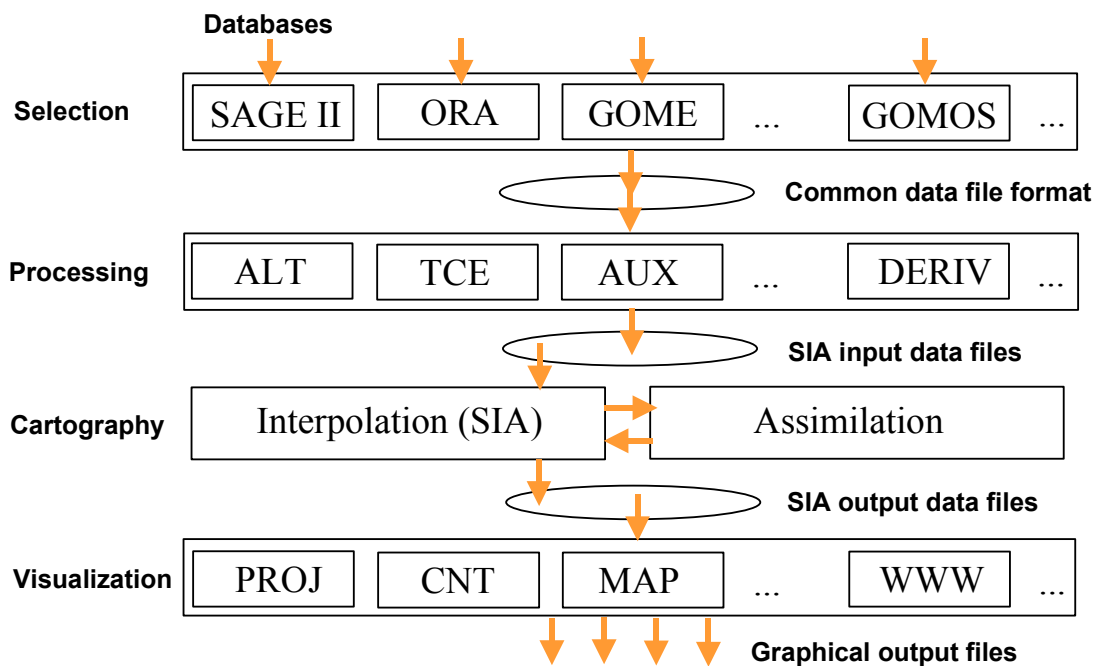


Fig. 3.1 Cartography software structure

The first layer constitutes an interface between the level-2 data as provided by the various sources and the common data file format adopted in this project (Appendix B.6). Each satellite instrument delivers data according to the characteristics of the experiment, resulting in a variety of data storage formats, type of information (profiles, total column, auxiliary data), differences in sampling, etc. By applying a translation into a common file format, the following layers become independent of the data source used. For each aerosol instrument we foresee a dedicated routine that handles the specifics of data selection and file format translation typical for that instrument. These specific selectors take as input user specified parameters, such as the time period for which data is wanted, the wavelength channel, etc. (section 3.2). At the moment the following interfaces are implemented:

- (i) SAGE II interface to our institute INGRES database
- (ii) ORA interface (only to total optical thickness flat file database)
- (iii) GOME interface to the total optical thickness flat file format as produced by the Prototypal GOME Aerosol Data Processor (PGADP) [Mochi et al., 1997a]

Adding new interfaces, for e.g. the future ENVISAT data products, is straight forward due to the highly modular structure of the software.

The second layer contains (simple) data processors, which produce user requested subsets from the data selected by the first layer. A first class of calculated data are auxiliary data. The following are available:

- (i) constant pressure surface
- (ii) constant temperature surface
- (iii) isentropic temperature surface
- (iv) tropopause heights from available temperature profiles by the WMO definition,
- (v) minimum valid altitude surface (due to presence of clouds)

A second type of processor calculates altitude intersections from the profiles, at a level determined by one of the above auxiliary surfaces.

Another useful processor is the extinction profile integrator, producing total column extinction values at the measurement points. The integration process starts at a low altitude limit, which can be:

- (i) tropopause height + 2 km (to avoid high altitude clouds)
- (ii) minimum valid altitude heights

Integration is carried out upwards to the highest available valid extinction value. Isolated missing or invalid points in the profile are replaced by an in altitude linear interpolated value.

Other processors are:

- (i) computation of zonal means of an altitude intersection
- (ii) zonal means of profiles, resulting in latitude-altitude intersections

Future extension foreseen and which will be located in this layer are:

- (i) time series production (e.g. animated data sets)
- (ii) production of wavelength dependent data

The output of this layer is in the format read by the interpolation algorithm. Extension of this layer with additional future processors should be straight forward.

The third layer handles the spherical interpolation/assimilation problem, which forms the core of the cartography. The interpolation is done entirely on the surface of the sphere, independent of any geographical projection chosen by the user afterwards. The interpolation interfaces with the assimilation because the assimilation is done on a regular 3D grid. The assimilation itself is an iterative process, requiring initial values at its regular grid. These values are provided by the interpolation. Final values, received back from the assimilation are further interpolated, because the assimilation grid is still too coarse for direct mapping. Interpolation and assimilation are totally separated from a software point of view. Data communication between both parts is done by file exchange. Cartography can be done with interpolation alone or using interpolation and assimilation.

The interpolation process depends on a number of parameters, such as interpolation grid resolution, choice between interpolation or fitting, local slope estimation parameters, smoothing filter parameters, etc. All parameters have default values or values which best suits a specific user application (see user profiles in the next section). The output of this layer is a set of files, which are to be read by the next visualisation layer.

The fourth layer does the graphical work and plots or prints the maps. The user can choose between different geographical projections, colour scales and contour options. Output is plotted on screen or stored in files with standard graphical formats, e.g. GIF, TIFF, JPEG, postscript, encapsulated postscript, etc. This makes it relatively simple to incorporate the results in publications, www servers, etc.

Except for the core of the interpolation algorithm, all routines are written in Interactive Data Language (IDL), (from Research Systems, Inc). Only the computation of the basis functions and the implementation of the truncated Taylor series interpolant are done in C. The C code is compiled into a sharable library, and routines herein are called from within IDL. The software runs and was tested on a HP UNIX workstation. Since IDL and C are available on almost any hardware platform, porting the software to any other system should not require any re-coding. The typical run time to produce a map with a one degree resolution in latitude and longitude, from 1000 sample points, is of the order of 10 seconds.

3.2 Description of parameters

The cartography process is controlled by a large set of parameters. These parameters define which actions are to be executed and what values are to be used. Parameters are provided to the system in a parameter file. This is an editable ASCII file, generated for a given cartography job and with values depending on the type of user. To ease the application of the system, the concept of user profiles have been introduced. A user profile is a description of the type of application for which a user needs aerosol data. For instance, an analyser of optical remote sensing data needs total column optical thickness values, for a certain area and a specific time, to correct its observations for the amount of sunlight absorbed by the aerosols in the atmosphere. A user profile is a shorthand for the set of parameters, required by the software, and which are given pre-set values in agreement with the user requirements. By defining various user profiles, the software becomes more user friendly and certain internal details, which could confuse the occasional user, remain hidden. At any time a user can customise its parameters.

The parameter file is an ASCII file having extension '.par'. It consists of an undetermined number of lines, each line being either a blank line or specifying a parameter.

A parameter specifying line has the following format:

```
KEYWORD = value ; comment
```

with:

```
KEYWORD      : name of the keyword
=             : obligatory separator
value        : value of the key, can be text or numbers
;            : end of data separator
comment      : user description of parameter (any free user text)
```

These words and delimiters can be separated by spaces and/or tabs for readability.

The order of the keyword lines in the file is unimportant.

Missing keywords are given default values ("", 0 or 0.0).

Keywords should be unique.

Additional keywords not used by the software at a given time may be present.

Below is given an overview of the parameters. They are divided in four groups, associated to the four software layers:

- (i) selection parameters
- (ii) processor parameters
- (iii) interpolation parameters
- (iv) visualisation parameters

The selection parameters control the data selection performed by the software. The following have been defined:

FILE= ../SAGEII_1020/TEST ; filepath (excluding extensions)

'../SAGEII_1020/' is the partial path referring to the directory where input and output files are stored.
'TEST' is the stem of the filenames read and generated by the software.

TITLE= Test ; a custom title

'Test' is the user title that will be put on the plot. If it is an empty string (just left blank), a title is composed automatically using the three following keywords.

DATABASE= SAGE II ; database name

'SAGE II' identifies the data.

CHANNEL= 1020 nm ; channel name

'1020 nm' further specifies the data.

SPECIES= Aerosols ; species name

'Aerosols' further specifies the product.

PERIOD= January 92 ; period description

'January 92' is the time frame put on the plot.

BEGIN_TIME= 19920101000000 ; begin of time frame

END_TIME= 19920131235959 ; end of time frame

The time selection of the data (YYYYMMDDhhmmss).

WAVELENGTH= 1020 ; in nanometer (nm)

FLAG_DUP= Y ; "Y"(es) or "N"(o) check for duplicate sample points

The interpolation algorithm cannot handle duplicate sample points. If Y, they will be removed before interpolation. Specify N if there are none or were removed previously.

FLAG_LOG= Y ; "Y"(es) or "N"(o) work with the logarithm of the data

If Y, the data read in will be converted to logarithmic scale. In this case, parameters referring to data values should be specified in logarithmic scale too.

FLAG_SF_SEL= N ; "Y"(es) or "N"(o) save selection files

The processor parameters control which processors are invoked and what values are to be used. The following have been defined:

MIN_ALT= 23.5 ; minimum altitude

MAX_ALT= 32.5 ; maximum altitude

ALTITUDE= 26.5 ; altitude where to make intersection (km)

PRESSURE= 50.4 ; pressure surface where to make intersection (mb)

TEMPERATURE= 200.0 ; temperature surface where to make intersection (degree K)

ISENTROPE= 500.0 ; isentropic temperature level where to make intersection (degree K)

FLAG_ALT= KM ; "KM", "MB" or "DK" altitude unit is in km, mb or degr K

FLAG_MVA= N ; "Y"(es) or "N"(o) minimum valid altitudes instead of tropopause

FLAG_UIG= N ; "Y"(es) or "N"(o) user supplied interpolation grid

FLAG_ASM= N ; "Y"(es) or "N"(o) assimilated data

FLAG_SF_PRE= N ; "Y"(es) or "N"(o) save processor files

The interpolation parameters control the data selection performed by the software. The following have been defined:

MIN_LON= -180.0 ; minimum longitude in degrees
MAX_LON= 180.0 ; maximum longitude in degrees
MIN_LAT= -90.0 ; minimum latitude in degrees
MAX_LAT= 90.0 ; maximum latitude in degrees

MIN_VAL= -8.0 ; valid range of sample data
MAX_VAL= 0.0 ; valid range of sample data

Both these values determine the valid scale.

BAD_VAL_MIN= -9.0 ; value to substitute for an underflowed bad sample
BAD_VAL_MAX= 1.0 ; value to substitute for an overflowed bad sample

Data values below MIN_VAL will be equalled to BAD_VAL_MIN.
Data values above MAX_VAL will be equalled to BAD_VAL_MAX.

FLAG_BAS= Y ; "Y"(es) or "N"(o) make basis functions
FLAG_INT= Y ; "Y"(es) or "N"(o) do interpolation
FLAG_BAD= N ; "Y"(es) or "N"(o) show bad points in map as holes

If Y, gaps will appear in the interpolation result. If N, bad points are removed and are not taken into account to compute the interpolation.

FLAG_CBP= Y ; "Y"(es) or "N"(o) correct bad point by interpolation of altitude profile

FLAG_ETP= N ; "Y"(es) or "N"(o) extrapolate to poles

The algorithm assumes the first sample point (index 0) to be the south pole (lon=0.0, lat=-90.0) and the last (index NR_SAMPLES - 1) the north pole (lon=0.0, lat=90.0). These points can be given a custom (possibly bad) value (when 'N'), or the algorithm can compute an extrapolation based on its Voronoi neighbours (when 'Y').

NR_LON_PTS= 360 ; number of longitudes to interpolate to
NR_LAT_PTS= 181 ; number of latitudes to interpolate to

Interpolation is to an angular equidistant, product grid.

DERIV_TYPE= 0 ; 0= no derivatives, use local fit, 1= user specified derivatives

The algorithm can be used in two modes. Usually no longitudinal or latitudinal derivative values are known, then put 0. If these derivatives are available then put 1. The former case solves the Lagrange interpolation problem, the latter case solves the Hermite interpolation problem.

NR_DER_LON= 0 ; number of longitudinal derivatives specified
NR_DER_LAT= 3 ; number of latitudinal derivatives specified

Functional values and derivative values are stored together in the file 'name.sam.val'. See the file description for details.

FIT_PAR= 2.4 ; parameter used in local thin plate fit

Valid range is ≥ 0.0 . If zero, then local interpolation is used instead of local fitting. The given value is a good default value. This interpolation or fit is used to estimate the derivatives in the neighbourhood of each of each sample point.

NR_FIT_PTS= -1 ; number of fit points (local set around a sample point)

Valid range -1, and >0 . This is the number of points used in the local interpolation or fit. This should be a number in the region 5 - 15. If entered -1, then the number of sample points used in the fit is taken equal to the number of Voronoi neighbours + 1 for that sample point. This is the most natural choice and gives the best results.

GFP_LON= 0.06 ; parameter in longitudinal gaussian filtering (1/e freq, in Nyquist freq units)

GFP_LAT= 0.12 ; parameter in latitudinal gaussian filtering (1/e freq, in Nyquist freq units)

After interpolation a final spatial low pass filtering can be done. These are the longitudinal and latitudinal characteristic normalised frequencies used in the Gaussian filter. Values should depend on the sampling rate. Good values for SAGE II: $0.03 \leq \text{GFP_LON} \leq 0.06$, $0.12 \leq \text{GFP_LAT} \leq 0.24$. The smaller the values, the smoother the resulting map.

FLAG_FLT_LON= Y ; "Y"(es) or "N"(o) final filtering in longitude

FLAG_FLT_LAT= N ; "Y"(es) or "N"(o) final filtering in latitude

FLAG_SF_BAS= Y ; "Y"(es) or "N"(o) save basis files (output of 'make_basis')

FLAG_SF_INT= N ; "Y"(es) or "N"(o) save interpolation files ('.sam' and '.int')

The above two flags control the deletion of the files produced during the run of the algorithm. Put FLAG_SF_BAS= Y if you need to do more than one interpolation for the same sample grid, but with different sample data values. This speeds up the calculations. Put FLAG_SF_INT= N if you're only interested in the picture outputs. Put FLAG_SF_INT= Y if you're interested in the interpolated values.

FLAG_ERR= N ; "Y"(es) or "N"(o) make error estimation for interpolant

The visualisation parameters control the data selection performed by the software. The following have been defined:

MIN_SCALE= -5.0 ; minimum of display scale

MAX_SCALE= -1.0 ; maximum of display scale

These both values determine the plot scale, usually contained within the valid scale.

FLAG_EPS= N ; "Y"(es) or "N"(o) print as an 'eps' file (else 'ps' file)

FLAG_SH_SAM= Y ; "Y"(es) or "N"(o) show window with sample points plot

FLAG_SH_INT= Y ; "Y"(es) or "N"(o) show window with interpolant plot

FLAG_SW_SAM= N ; "Y"(es) or "N"(o) save window with samples in .gif file

FLAG_SW_INT= N ; "Y"(es) or "N"(o) save window with interpolant in .gif file

FLAG_PR_SAM= N ; "Y"(es) or "N"(o) print sample points to .ps file

FLAG_PR_INT= N ; "Y"(es) or "N"(o) print interpolation to .ps file

3.3 Limitations of the interpolation algorithm

The interpolation algorithm was found to be numerically very stable and robust. Many tests were done with the number of sample points varying from a few tens up to 10000. The highly irregular sample grids, encountered in the SAGE II, ORA and GOME experiments, never caused the algorithm to fail. Also the derivative estimation, based on a local thin plate shape estimation of the variation of the data, never showed signs of ill-conditioning.

The limits of the cartography based on interpolation alone are due to the lack of coverage of the data, rather than the algorithm itself. Even in cases where no data is available over large areas of the globe, an estimate is produced by the algorithm in these areas, based on the trend of the data at the boundary of these areas. It is clear that the interpolated values in these areas are less trustworthy, as compared to the more densely sampled areas. Data sets showing limited coverage are better not mapped based on interpolation alone and require assimilation to fill in the gaps.

The current implementation of the interpolation algorithm can use the errors on the data values as weights in the local fitting procedure. Up to now this is the only place where error information is used. As a possible extension of the algorithm, one could consider using certain ideas from statistical interpolation to improve the quality of the interpolant for data with important error bars. For instance, one could consider a modification of the local fitting method, to minimise the interpolation error or covariance of the interpolated values.

3.4 Pseudo code description of the software

Below is given a high-level pseudo-code description of the core of the interpolation algorithm.

A. Basis functions

To each sample point is associated a unique basis function. All basis functions are tabulated in each interpolation point. In practice, since we use local basis functions (having compact support over only the Voronoi polygon), basis functions only have to be evaluated inside their Voronoi polygon, for they are defined to vanish outside. Basis functions are computed by applying a one-dimensional construction over geodesics passing through the central sample point of the Voronoi polygon. This radial construction involves a w-function and a slope function. The latter is independent of the sample points. The tabulation of basis functions only needs as input the locations of the sample points, the interpolation grid and information about the connectivity of the Delaunay spherical triangulation. This part of the interpolation algorithm is therefore independent of the functional data.

tabulate basis functions

```

/*
 * Tabulate all basis functions associated with the sample points
 * in the interpolation points on the unit sphere.
 *
 * Each function is of finite support and vanishes identically outside
 * the spherical Voronoi polygone associated with its point.
 * The basis function associated to a sample point is
 * tabulated in those interpolation points which are in its support area.
 *
 * Each interpolation point belongs to a unique triangle.
 * Each triangle belongs to exactly three Voronoi polygons
 * (those having as centre points the vertices of the triangle).
 * So there will always be three contributing basis functions to
 * a given interpolation point.
 */
{
    /*
     * Construction of auxiliary data related to the triangular decomposition
     */
    make Voronoi data
    make interpolation grid vectors
    make sample grid vectors
    make influence list

```

```

/*
* Basis function computation
*/
calculate phi functions
calculate basis functions

/*
* Destruction of auxiliary data
*/
cleanup Voronoi data
cleanup interpolation grid vectors
cleanup sample grid vectors
cleanup influence list
}

```

calculate phi functions

```

/*
* Construct fundamental (phi) functions with compact support over
* a single Voronoi polygon.
* They are one at the centre point and zero at the polygon boundary.
* Radial construction over geodesics through the local sample point.
*/
{
  loop over interpolation points
  {
    loop over influencing triangle vortexes
    {
      From vertex and interpolation point calculate geodesic.
      Determine intersections of geodesic with the Voronoi polygon boundary.
      Evaluate w function over geodesic in interpolation point.
      Compute value of the slope function from w value.
      Assign results as phi value.
    }
  }
}

```

calculate basis functions

```

/*
* Scale fundamental phi functions to basis functions to obtain a partition of unity.
*/
{
  loop over interpolation points
  {
    loop over influencing triangle vertexes
    {
      sum the three phi values for the considered interpolation point,
      basis function value = phi function value/sum.
    }
  }
}

```

B. The interpolant

The interpolation formula can be used in two ways. First, directly when derivative values are available. Secondly, by providing only functional values, in which case the algorithm estimates all local

derivatives from the determination of an interpolating or fitting thin plate shape. The computation of the local fit shape around each sample point, requires the solution of a (low dimensional) linear system. The size of this system depends on the correlation length one assumes for the data at hand. Usually only nearest neighbours (i.e. Voronoi boundary samples) are used to determine the local shape, resulting in linear system dimensions of the order of 10.

```
interpolate
```

```
{
    make fits
    make interpolant
    cleanup
}
```

```
make fits
```

```
/*
 * For each sample point a local thin plate shape interpolating or fitting the data is computed.
 */
{
    loop over sample points
    {
        Make local linear system associated with the local thin plate fit,
        using functional values in a neighbourhood of the sample point.
        Solve local linear system.
        Store local thin plate shape coefficients.
    }
}
```

```
make interpolant
```

```
/*
 * Tabulate the interpolant in each interpolation point on the unit sphere.
 * Each interpolant is influenced by three local thin plate fit functions,
 * weighted with three zero order basis functions.
 */
{
    loop over the interpolation points
    {
        loop over the contributing triangle vertexes
        {
            Get for basis function associated to vertex its value in the interpolation point.
            Evaluate the thin plate shape associated to vertex in the interpolation point.
            Compute interpolant value.
        }
    }
}
```

3.5 Algorithm validation

3.5.1 Interpolation of a known function

As test function we consider a distribution which is Gaussian in latitude and uniform over longitudes

$$f(\phi, \theta) = a \exp(-(\theta/\theta_c)^2) + b \quad (5.1.1)$$

This is a rough approximation to a typical aerosol distribution in the stratosphere. Typical parameter values are $a = 8$, $b = -8$, $\theta_c = 57^\circ$ (base 10 logarithmic scale). This function was chosen because it is a rough approximation to the typical aerosol distribution at a fixed geometric altitude. The sample grid is shown in Fig. 5.1.

Root mean square error is defined as:

$$\epsilon_{RMS} = \sqrt{\frac{\sum_{n=1}^N (\tilde{f}_n - f_n)^2}{\sum_{n=1}^N (f_n)^2}} \quad (5.1.2)$$

wherein f_n, \tilde{f}_n are the exact and interpolated functional values at the interpolation grid points.

a) Functional values only, local thin plate interpolation (Fig. 5.2)

Maximum absolute error: 0.227
 Maximum relative error: 0.051
 Mean absolute error: 0.051
 Mean relative error: 0.012
 Root mean square error: 0.016

b) Functional values and first order gradients, truncated MT series

Maximum absolute error: 0.386
 Maximum relative error: 0.087
 Mean absolute error: 0.070
 Mean relative error: 0.016
 Root mean square error: 0.022

c) Functional values, first and second order derivatives, truncated MT series

Maximum absolute error: 0.042
 Maximum relative error: 0.0096
 Mean absolute error: 0.0063
 Mean relative error: 0.0014
 Root mean square error: 0.0021

d) Functional values, first, second and third order derivatives, truncated MT series (Fig. 5.3)

Maximum absolute error: 0.014
 Maximum relative error: 0.0033
 Mean absolute error: 0.0018
 Mean relative error: 0.00040
 Root mean square error: 0.00061

The cases b) to d) illustrate the convergence of the MT series representation for increasing number of given (exact) derivatives. It also shows that, at least for this test case, the thin plate derivative

estimation does better than giving exact first derivatives, but is worse than given exact first and second order derivatives.

3.5.2 Interpolation error caused by uncertainties on the data

We will now give an expression for the error on the interpolant when applied to measurement values f_n subject to uncertainties ϵ_n

$$f_n = \hat{f}_n \pm \epsilon_n \quad (5.2.1)$$

Substitution of the expressions for the local thin plate shapes (B.4.5) in the interpolant (B.4.10) yields

$$A[f](\phi, \theta) = \sum_{n=1}^N \left(\sum_{i=1}^{I_n+3} a_i K_i(\phi, \theta) \right) U_n^{0,0}(\phi, \theta) \quad (5.2.2)$$

The local coefficients a_i are the solution of the local linear system (B.4.7), hence

$$A[f](\phi, \theta) = \sum_{n=1}^N \sum_{j=1}^{I_n} \left(\sum_{i=1}^{I_n+3} [\mathbf{M}^{-1}]_{ij} K_i(\phi, \theta) \right) f_{m(j)} U_n^{0,0}(\phi, \theta) \quad (5.2.3)$$

Define

$$L_j(\phi, \theta) = \sum_{i=1}^{I_n+3} K_i(\phi, \theta) [\mathbf{M}^{-1}]_{ij} \quad (5.2.4)$$

Then

$$A[f](\phi, \theta) = \sum_{n=1}^N \sum_{j=1}^{I_n} f_{m(j)} L_j(\phi, \theta) U_n^{0,0}(\phi, \theta) \quad (5.2.5)$$

In the second summation is j a local index and runs over the Voronoi samples associated to the n -th sample. With $m(j)$ we denote the map (depending on n) from local to global sample point numbering.

Due to the linearity of the interpolant we get the error function

$$\epsilon[A[f]](\phi, \theta) = \sum_{n=1}^N \sum_{j=1}^{I_n} \epsilon_{m(j)} L_j(\phi, \theta) U_n^{0,0}(\phi, \theta) \quad (5.2.6)$$

which gives the error on the interpolant caused by the uncertainties on the sample data.

We apply this error computation to the SAGE II measurements for JAN 92 at an altitude of 26.5 km. The sample values have the following error statistics:

Maximum absolute error:	2.896
Maximum relative error:	0.989
Mean absolute error:	0.097
Mean relative error:	0.033
Root mean square error:	0.081

We get the following error statistics on the interpolated values (thin plate fit):

Maximum absolute error:	2.771
Maximum relative error:	0.958
Mean absolute error:	0.104
Mean relative error:	0.036
Root mean square error:	0.070

This example shows that the mean relative error of the interpolated values, (3.6%), are of the same order as the mean relative errors of the samples, (3.3%). The interpolation algorithm does not significantly adds to the uncertainty of the interpolated data. Statistical interpolation techniques could reduce the uncertainty on the interpolated values. This is an option for a future improvement of the algorithm.

3.5.3 Self consistency of the interpolation

A test has been devised to measure the self-consistency of the interpolant. It consists of removing a single sample point from the set of samples and to let the interpolation algorithm reproduce this sample value from the remaining. This is done in succession for each sample point and the resulting errors were averaged.

We applied this to the Gaussian test distribution (5.1.1), with sample points as shown in Fig. 5.1 and found the following error statistics:

Maximum absolute error:	0.181
Maximum relative error:	0.043
Mean absolute error:	0.067
Mean relative error:	0.016
Root mean square error:	0.019

We find the consistency error to be comparable to the interpolation error as obtained from the case a) in section 5.1, which was to be expected.

3.6 Examples of aerosol maps from cartography

3.6.1 SAGE II

The data used for the examples shown in this section were obtained from the Stratospheric Aerosol and Gas Experiment II (SAGE II) instrument on board the American Earth Radiation Budget Satellite (ERBS). The SAGE II instrument is of the limb scanning, solar occultation type. The data has the following features.

- (i) Geographic coverage, typical 60°S – 60°N, maximum 80°S – 80°N.
- (ii) Time coverage from 10/84 till now.
- (iii) Wavelength channels: 0.385, 0.485, 0.525, 1.02 μm .
- (iv) Altitude profiles, typical from 15 – 50 km, with 1km resolution.
- (v) Provided quantity: beta = aerosol optical extinction coefficient, (1/km).
- (vi) Data type: validated level-2

3.6.1.1 Altitude sections for January 1992 (post Pinatubo period)

Figures 6.1.1 – 6.1.10 show aerosol maps in the altitude region 23.5 – 32.5 km made from the January 1992 profiles and for the 1.02 μm wavelength channel. All figures use the same colour scale. We are about 6 months after the eruption of the Pinatubo volcano (June 1991). Large quantities of dust were ejected into the stratosphere. At equatorial latitudes the amount of sunlight penetrating through the atmosphere then fell below the instrument sensitivity and no valid data are available over large portions of the globe. These figures show that the majority of the dust is concentrated at the lower stratosphere.

Any invalid values, falling outside user defined limits, are rejected by the algorithm. The resulting (even large) gaps are no threat to the robustness of the algorithm, but will of course reduce the trustfulness of the map. Profiles with invalid values are linearly interpolated in altitude, if possible, to reduce the number of gaps in a map. The user can choose between keeping invalid samples in the map (and giving them a custom bad value, showing up as a ‘hole’ in the map) or rejecting the sample (so that it does not disturb the interpolation).

3.6.1.2 Altitude sections for January 1989 (pre Pinatubo period)

Figures 6.1.11 – 6.1.15 show aerosol maps for the 1.02 μm wavelength channel at the five pressure levels used by the assimilation algorithm, 102.5, 73.4, 50.4, 30.0 and 10.0 mbar, for January 1989. All figures use the same colour scale. January 1989 fell in a volcanic calm period. As can be seen from comparison with January 1992, the aerosol level is about two orders of magnitude lower.

3.6.1.3 Stratospheric variability in 1992 at 26.5 km altitude

Figures 6.1.16 – 6.1.27 show a) measurement points and b) aerosol maps at a fixed altitude of 26.5 km, for each month of the year 1992 and for the 1.02 μm wavelength channel. All figures use the same colour scale. One notices the decay of the mean aerosol level from January to December, due to precipitation of the volcanic aerosols. The point plots reveal the lack of coverage by the SAGE II instrument, in equatorial regions for certain periods of the year (February, July), due to the inclination of the satellite orbit. The equatorial aerosol reservoir is not sampled for these periods and consequently the corresponding maps are incomplete. However, these periods are a good test of the algorithm, showing that large under-sampled areas together with other densely sampled areas still produce a reasonable map, in agreement with what to expect from the trend of the sample values.

In order to produce trustful maps for problem months, such as February and July, a combination of assimilation and interpolation should be used.

3.6.1.4 Total column optical thickness during the period August 15 – September 24, 1993

Figure 6.1.28 show a) measurement points and b) a total column optical thickness map for the period August 15 till September 24, 1993. The total column value was obtained by integrating the SAGE II profiles from the tropopause + 2 km level upward to the last available valid value. The definition for the tropopause was taken from the WMO.

The obtained map is in good agreement with the one published by Trepte et. al. [1994], Fig. 1a. Both maps are for the 1.02 μm wavelength channel. However, some regular blob-like features can be seen near the south pole in Fig. 1a. They are probably artefacts of the interpolation method used by the authors. These features are absent in our map, Figure 6.1.28b.

3.6.2 ORA

The data used for the examples shown in this section were obtained from Occultation Radiometer (ORA) instrument, flown on board the European Retrieval Carrier mission (EURECA). The ORA instrument is of the limb scanning, solar occultation type. The data has the following features.

- (i) Geographic coverage, typical 40°S – 40°N.
- (ii) Time coverage from 11/08/92 till 13/05/93.
- (iii) Wavelength channels: 0.259, 0.340, 0.385, 0.435, 0.442, 0.525, 0.600, 0.943, 1.013 μm .
- (iv) Altitude profiles, typical from 15 – 60 km, with 2 km resolution.
- (v) Provided quantity: beta = aerosol optical extinction coefficient, (1/km).
- (vi) Data type used: total aerosol optical extinction at 1.013 μm .

3.6.2.1 Total column optical thickness during the period September 1992 – April 1993

Figures 6.2.1-6.2.8 show a) measurement points and b) total column optical thickness maps for the months September 1992 till April 1993 and for the 1.02 micron wavelength channel. The total column value was obtained by integrating the ORA profiles from the tropopause + 2 km level upward to the last available valid value and by using the WMO tropopause definition. The ORA experiment had a smaller latitudinal coverage than the SAGE II experiment and was limited in time to about the above period.

3.6.3 GOME

The data used for the examples shown in this section were obtained from Global Ozone Monitoring Experiment (GOME) instrument, on board the European Earth Remote Sensing satellite 2 (ERS-2). The GOME instrument is a nadir looking spectrometer, measuring solar radiation scattering. The data has the following features.

- (i) Geographic coverage: global.
- (ii) Time coverage from 01/96 till now.
- (iii) Wavelength range: from 0.240 – 0.790 μm .
- (iv) Provided quantity: total column aerosol optical extinction.
- (v) Processing of the data is still ongoing.
- (vi) Data type used: PGADP aerosol test products.

3.6.3.1 Total column optical thickness

Figs. 6.3.1a and b shows the global sample point distribution and the mapped aerosol optical thickness respectively, over sea, for the period 11-07-1997 till 14-07-1997. The data selection was made with the following parameters (applied in logical AND) [Mochi et al., 1997b]:

- (i) PMD cloud flag = 0 (no cloud covering in pixel)
- (ii) sun zenith angle ≤ 70 degrees
- (iii) total aerosol optical thickness between 0 and 1.5

Because the PGADP version, used to produce the GOME data used here, had difficulties in producing reliable cloud flags over land, these values were suppressed in the map.

A similar result is shown in Figs. 6.3.2a and b for the period 14-07-1997 till 16-07-1997 (no time overlap with the previous period). The same data selection parameters were used.

4 Assimilation of stratospheric aerosols using remote sensing optical data.

In this work the adjoint model technique will be extensively used, therefore we give a brief theoretical overview of adjoint models and their application in atmospheric composition data assimilation, followed with the implementation for the aerosol assimilation.

4.1 Introduction to adjoint models

Adjoint models are tools developed for inverse modelling of physical systems. Inverse modelling is used in various fields of science such as geophysics and molecular physics. Among the applications of adjoint models in oceanography and meteorology are data assimilation, model tuning, sensitivity analysis, and determination of singular vectors.

In atmospheric sciences, combining a model with data is a crucial task. Several methods have been developed for data assimilation. Sequential methods put a model in a state, which in general is not consistent with its dynamics and/or physics. The model is disturbed and needs some time to reach consistency. To keep the disturbance as small as possible, data are prepared and only a correction of the model state “in the direction” of the data takes place. Several sequential methods differ in the degree of consistency of the correction with model dynamics and/or physics, e.g. Nudging, Successive Correction, Optimal Interpolation, Kalman Filtering. In contrast, the adjoint method always guarantees full consistency with the dynamics and/or physics. By variation of control variables it is intended to adjust a model trajectory as close as possible to the data. To quantify the misfit of a model prediction, a cost function is introduced. This cost function is minimised by use of an iterative algorithm. Starting with a first guess, in each iteration step an improved vector of control variables is searched. Thereby the search direction is computed from the gradient of the cost function with respect to the control variables. The adjoint model computes this gradient vector. In data assimilation, the control variables typically determine the initial conditions or the forcing for the model [Talagrand and Courtier, 1987]. The use of an adjoint model in an optimisation procedure is described in the following section.

4.1.1 Adjoint models and the assimilation concept

Consider a dynamical physical system and a model describing this system. Let Y be a set of observations and suppose that the model can compute the values S corresponding to these observations. How can the model be manipulated in order to obtain an optimal fit between observations and corresponding model values?

To quantify the misfit we introduce a cost function

$$J = \frac{1}{2}(S - Y, S - Y) \quad (1)$$

By the choice of an appropriate inner product (\cdot, \cdot) . This implies that least-squares-fitting is intended. The smaller J is the better the model fits the data.

In order to manipulate the model, we specify a set of n parameters X , which are called control variables in the following. The dependence of S on X within the model is given by a mapping

$$\begin{aligned} \mathbf{H} : R^n &\rightarrow R^m \\ X &\rightarrow S \end{aligned} \quad (2)$$

Thus, J can be expressed in terms of X by

$$\begin{aligned}
J : R^n &\rightarrow R \\
X &\rightarrow \frac{1}{2}(\mathbf{H}(X) - Y, \mathbf{H}(X) - Y)
\end{aligned} \tag{3}$$

The problem is to determine the set of control variables X that minimises J . Effective minimisation algorithms require that the gradient $\nabla_X J(X_0)$ of J with respect to the control variables at a given point X_0 . To first order we write the Taylor expansion of J

$$J(X) = J(X_0) + (\nabla_X J(X_0), X - X_0) + o(|X - X_0|) \tag{4}$$

Or, in short terms,

$$\delta J = (\nabla_X J(X_0), \delta X) \tag{5}$$

In the following we will use this shorthand notation whenever linear approximations are involved. Suppose \mathbf{H} is sufficiently regular, then for each parameter vector X_0 , a variation of Y can be approximated to the first order by

$$\delta Y = \mathbf{A}(X_0)\delta X, \tag{6}$$

Where $\mathbf{A}(X_0)$ denotes the Jacobian of \mathbf{H} at X_0 . Due to the symmetry of the inner product and the product rule the differentiation of (3) yields

$$\begin{aligned}
\delta J &= \frac{1}{2}(\mathbf{A}(X_0)\delta X, \mathbf{H}(X_0) - Y) + (\mathbf{H}(X_0) - Y, \mathbf{A}(X_0)\delta X) \\
&= (\mathbf{H}(X_0) - Y, \mathbf{A}(X_0)\delta X)
\end{aligned} \tag{7}$$

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

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

we obtain

$$\delta J = (\mathbf{A}^*(X_0)(\mathbf{H}(X_0) - Y), \delta X). \tag{9}$$

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

$$\nabla_X J(X_0) = \mathbf{A}^*(X_0)(\mathbf{H}(X_0) - Y). \tag{10}$$

The linear operator $\mathbf{A}(X_0)$ represents the tangent linear model. Its adjoint $\mathbf{A}^*(X_0)$, which is linear as well, represents the adjoint model. Both operators depend on the point X_0 at which the model is linearised. According to (10) the difference $\mathbf{H}(X_0) - Y$ can be interpreted as a forcing of the adjoint model. A detailed analysis of required basic numerical operations yields that the computation of the cost function and its gradient for a given vector of control variables takes only 2-5 times the computation of the cost function. Alternatively, the gradient vector $\nabla_X J(X_0)$ could be approximated by finite differences, which needs at least $n+1$ computations of the cost function. The use of the adjoint modal has two advantages compared to finite differences. First, it saves a lot of run time, especially for large n , and secondly, the computed gradient is exact.

4.1.2 A discretised application: 4D-Variational data assimilation on atmospheric composition

Based on the previous theoretical framework four-dimensional, variational data assimilation (4D-Var) can be formulated. 4D-Var seeks to produce an analysis, which fits a set of observations taken over a period of time, subject to the strong constraint that the evolution of the analysed quantities is governed by a deterministic model. By imposing the equations of the model as strong constraints, the analysis problem is reduced to that of determining initial values for the model such that the subsequent evolution minimises a measure of the fit to the observations.

4D-Var expresses the analysis problem as the minimisation of a cost function, J , defined as

$$J = \frac{1}{2}(\mathbf{x}_b - \mathbf{x}_0)^T \mathbf{B}^{-1}(\mathbf{x}_b - \mathbf{x}_0) + \frac{1}{2} \sum_{n=0}^N (\mathbf{y}_n - \mathbf{s}_n)^T \mathbf{R}_n^{-1}(\mathbf{y}_n - \mathbf{s}_n). \quad (11)$$

Here, \mathbf{x}_0 is the vector of initial parcel concentrations, \mathbf{x}_b is an independent estimate of the initial parcel concentrations, and \mathbf{B} is the covariance matrix of expected errors in \mathbf{x}_b . The expression $(\mathbf{x}_b - \mathbf{x}_0)^T \mathbf{B}^{-1}(\mathbf{x}_b - \mathbf{x}_0)$ is generally called the ‘background term’ of the cost functional, and \mathbf{x}_b is called the background.

The vector \mathbf{y}_n in Eq. (11) consists of all observations which are considered valid at the timestep n ; \mathbf{s}_n is a vector of ‘model equivalents’ of the observations. That is, each element of \mathbf{s}_n is an estimate of the corresponding element of \mathbf{y}_n , based, in this application of the method, on the parcel concentrations at time step n . In the analyses we present here, \mathbf{s}_n is a linear function of parcel concentrations, i.e.

$$\mathbf{s}_n = \mathbf{H}_n \mathbf{x}_n, \quad (12)$$

where \mathbf{x}_n is the vector of concentrations of all species for all parcels at timestep n , and \mathbf{H}_n is an ‘observation operator’.

The matrix \mathbf{R}_n is the covariance matrix for the random errors in the term $(\mathbf{y}_n - \mathbf{s}_n)$ which would be expected given a perfect analysis. That is, \mathbf{R}_n accounts for the random errors in the observations and the ‘representativeness errors’ introduced in simulating the observations.

The strong constraints of the model equations are incorporated into the analysis by regarding J as a function of the initial concentrations only, i.e. as a function of \mathbf{x}_0 . Concentrations at subsequent times are determined by integrating the model equations forward in time. This procedure produces two major simplifications. First, it replaces a constrained minimisation problem with an unconstrained problem. (Numerical algorithms for unconstrained minimisation are considerably more efficient and less prone to problems of ill-conditioning than are algorithms for constrained minimisation.) Second, the number of independent variables is reduced by a factor of $N+1$.

The analysis scheme uses a descent algorithm to produce a convergent sequence of estimates of the vector \mathbf{x}_0 which minimises the cost functional. The algorithm requires the calculation of the gradient of the cost function with respect to \mathbf{x}_0 . This is evaluated by integrating the adjoint of the model. These equations may be derived in a number of ways.

Define the functional J_m in the form

$$J_m = \frac{1}{2} \sum_{n=m}^N (\mathbf{y}_n - \mathbf{s}_n)^T \mathbf{R}_n^{-1}(\mathbf{y}_n - \mathbf{s}_n). \quad (13)$$

Now consider an infinitesimal variation, $\delta\mathbf{x}_0$, in the initial concentrations. At each subsequent step m of the model, there will be corresponding infinitesimal variations, $\delta\mathbf{x}_m$ and δJ_m in the concentrations and in the functional J_m .

J_m depends only on concentrations at step m and later. Since these concentrations are uniquely determined by the equations of the model and the concentrations for any step $i \leq m$, it is legitimate to regard J_m as a function of the concentrations at step i only. By definition, the gradient of J_m with respect to the concentrations at step i satisfies the equation

$$\delta J_m = (\nabla_{\mathbf{x}_i} J_m)^T \delta \mathbf{x}_i. \quad (14)$$

Suppose that $\nabla_{\mathbf{x}_m} J_m$ is known. We wish to calculate $\nabla_{\mathbf{x}_{m-1}} J_{m-1}$, and hence, by induction to calculate

$$\nabla_{\mathbf{x}_0} J = \nabla_{\mathbf{x}_0} J_0 + \nabla_{\mathbf{x}_0} \frac{1}{2} \{(\mathbf{x}_b - \mathbf{x}_0)^T \mathbf{B}^{-1} (\mathbf{x}_b - \mathbf{x}_0)\}. \quad (15)$$

From the definition J_m , we have

$$\nabla_{\mathbf{x}_{m-1}} J_{m-1} = \nabla_{\mathbf{x}_{m-1}} J_m + \frac{1}{2} \nabla_{\mathbf{x}_{m-1}} \{(\mathbf{y}_{m-1} - \mathbf{s}_{m-1})^T \mathbf{R}_{m-1}^{-1} (\mathbf{y}_{m-1} - \mathbf{s}_{m-1})\}. \quad (16)$$

The ease with which the second term in Eq. (16) may be evaluated depends on the complexity of the method of simulating observations. To evaluate the first term in Eq. (16), note that for an arbitrary infinitesimal variation in \mathbf{x}_{m-1} , we have by Eq. (14),

$$\delta J_m = (\nabla_{\mathbf{x}_{m-1}} J_m)^T \delta \mathbf{x}_{m-1}$$

and

$$\delta J_m = (\nabla_{\mathbf{x}_m} J_m)^T \delta \mathbf{x}_m.$$

Hence

$$(\nabla_{\mathbf{x}_{m-1}} J_m)^T \delta \mathbf{x}_{m-1} = (\nabla_{\mathbf{x}_m} J_m)^T \delta \mathbf{x}_m. \quad (17)$$

If the equations of the model are written as

$$x_m = \mathbf{M}_{m-1}(x_{m-1}) \quad (18)$$

then, for infinitesimal variations,

$$\delta x_m = \mathbf{M}_{m-1} \delta x_{m-1} \quad (19)$$

where \mathbf{M}_{m-1} is a matrix whose elements are the partial derivatives of the elements of \mathbf{M}_{m-1} with respect to the elements of \mathbf{x}_{m-1} .

Substituting Eq. (19) into Eq. (17) gives

$$\begin{aligned} (\nabla_{\mathbf{x}_{m-1}} J_m)^T \delta \mathbf{x}_{m-1} &= (\nabla_{\mathbf{x}_m} J_m)^T (\mathbf{M}_{m-1} \delta \mathbf{x}_{m-1}) \\ &= (\mathbf{M}_{m-1}^T \nabla_{\mathbf{x}_m} J_m)^T \delta \mathbf{x}_{m-1} \end{aligned} \quad (20)$$

Since this equation holds for arbitrary $\delta \mathbf{x}_{m-1}$, we must have

$$\nabla_{\mathbf{x}_{m-1}} J_m = \mathbf{M}_{m-1}^T \nabla_{\mathbf{x}_m} J_m. \quad (21)$$

Hence, by Eq. (16),

$$\nabla_{\mathbf{x}_{m-1}} J_{m-1} = \mathbf{M}_{m-1}^T \nabla_{\mathbf{x}_m} J_m + \frac{1}{2} \nabla_{\mathbf{x}_{m-1}} \{(\mathbf{y}_{m-1} - \mathbf{s}_{m-1})^T \mathbf{R}_{m-1}^{-1} (\mathbf{y}_{m-1} - \mathbf{s}_{m-1})\}. \quad (22)$$

This is the adjoint tangent linear (ATL) equation. Given $\nabla_{\mathbf{x}_m} J_m$, the equation allows $\nabla_{\mathbf{x}_{m-1}} J_{m-1}$ and so by induction, $\nabla_{\mathbf{x}_0} J_0$ to be calculated. Once $\nabla_{\mathbf{x}_0} J_0$ is known, the required gradient of the cost function with respect to the initial conditions is given by Eq. (15). Starting the induction at step N we have

$$\nabla_{\mathbf{x}_N} J_N = \frac{1}{2} \nabla_{\mathbf{x}_N} \{(\mathbf{y}_N - \mathbf{s}_N)^T \mathbf{R}_N^{-1} (\mathbf{y}_N - \mathbf{s}_N)\}. \quad (23)$$

The algorithm used to minimise J is as follows.

1. Start with an initial guess for \mathbf{x}_0 .
2. Integrate the model to give \mathbf{x}_n , for $n=1, \dots, N$.
3. Evaluate J . If the value of J is small enough STOP.
4. Iterate the ATL equations to calculate $\nabla_{\mathbf{x}_0} J$.
5. Use a descent algorithm to find a better guess at \mathbf{x}_0 (i.e. a guess for which J is reduced).
6. GOTO 2

4.1.3 Development of the adjoint model core

The adjoint model plays a very important role in data assimilation. Since in the context of this project various models with a different degree of complexity of aerosols will be used in the process of assimilation, the development of the adjoint model code must be reliable, easy to maintain, compliant with programming standards and bug free.

Several methods exist to develop adjoint model code. In order to have a general standardised procedure for the development of the adjoint model code, we choose to adopt the use of the Adjoint Model Compiler (AMC) [Giering, 1992]. This is a source transformation tool, it is based on the idea that a numerical model is an algorithm that can be viewed as a composition of differentiable functions, each representing a statement in the numerical code. The algorithm imposes the order of evaluation of the individual functions. Differentiation of the composition can be done by applying the chain rule. According to how the chain rule is used to propagate through the computation, one can distinguish two approaches to Automatic Differentiation (AD): the ‘forward’ and ‘reverse’ modes, leading to tangent linear model and adjoint linear model. For initial verification purposes, we also will develop ‘pseudo’-adjoint code with the automatic differentiation tool ADIFOR [Bischof, 1992].

The use of these tools allows to set-up a general procedure for the development and maintenance of assimilation code, based on existing physical models. This allows a general procedure to maintain 4D-Var code limited to the maintenance of the “physical model” code.

The general procedure for the 4D-Var code generation has been laid down as described in the previous section. The main advantages of this general procedure are the following.

- Automatic code generation of the necessary adjoint model;
- Generated adjoint code is compliant with programming standards, reduced risk for errors, and optimised;
- Easy maintenance of code, only the dynamical and physical models need to be maintained;
- Efficient implementation of new physical models.

Implementation of AD tools

Currently, two AD tools have been implemented; ADIFOR and AMC. A simple test case was used to test the implementation of these AD tools for the purpose of assimilation.

Description of test case.

Consider two species S and R, which evolve in time as

$$\frac{dS}{dt} = -b \times S \quad \text{with } S = 2.0 \text{ at } t = 0$$
$$\frac{dR}{dt} = b \times S \quad \text{with } R = 0.0 \text{ at } t = 0$$

As an assimilation example, we use 4 observations of S at 4 different times. These observations are of the analytical solution of the evolution of S. The purpose of the exercise is to assimilate corresponding values of R, without having observations of R, and relying on the coupling of S and R. Initial guesses for the values at $t = 0$ are $S = 1.0$ and $R = 1.0$. The matrices \mathbf{H} and \mathbf{R}_n are taken identity matrices. The following figures (figure wp1200.1) give the evolution of the cost function in the iterative process, and the time evolution of R for different iterations, the full line being the 5th iteration.

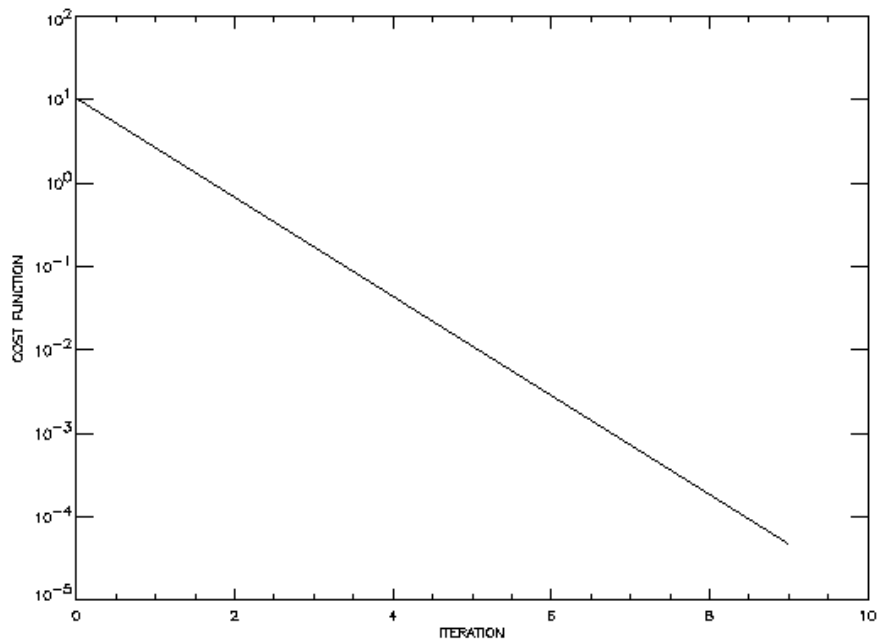
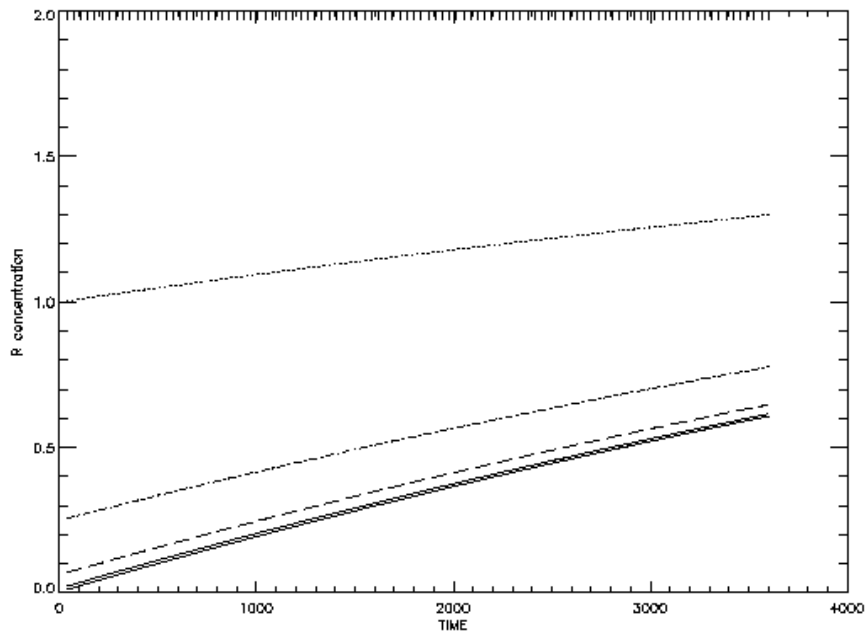


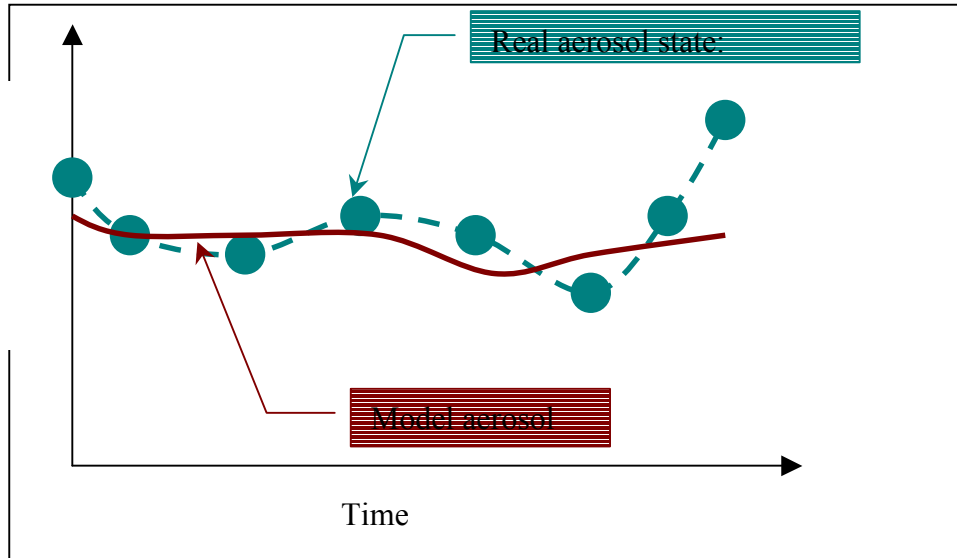
Figure wp1200.1: see text.

4.2 Introduction to 4D-Var aerosol assimilation.

The assimilation of stratospheric composition in general, and aerosols in particular, is a new concept. This concept has been since long recognised in the numerical weather prediction community, where big

improvements in the weather forecasts have been obtained by improving the assimilation technique. The latest improvement, namely a full 4D-Var assimilation technique, has proven to be a milestone in numerical weather prediction. The application of the 4D-Var technique to stratospheric aerosols is thus new. Results currently obtained, are only indicative of what can be obtained with 4D-Var assimilation if improvements are allowed.

As a reminder the following schematic picture of 4D-Var is given:



4D-Var aims at finding an initial condition x_0 for the model which minimises the distance between model and observations within a given period to obtain an optimal fit between the observations and corresponding model values. This is equivalent to finding a model initial condition which produces a model trajectory as close as possible to the real trajectory for a given period. The distance or misfit between model values and observations is referred to as the “cost”.

The result or output of the 4D-Var assimilation is thus an initial condition for the forward model, the real state of the aerosols can be “reproduced” by running the forward model over the assimilation time period.

4.2.1 Mono-wavelength approach.

Stratospheric aerosol data are mainly inferred from by remote sensing optical data. Using optical extinction at various wavelengths, valuable information of the characteristics of the stratospheric aerosols can be retrieved. However, in this project, a mono-wavelength approach is discussed and implemented. In contrast to this simple set-up, a multi-wavelength approach would require the following knowledge:

- A model to convert aerosol extinction data at different wavelengths into size distribution and composition of stratospheric aerosols in all conditions,
- A microphysical model that describes the formation and evolution of the aerosol size distribution,
- The temporal and spatial distribution and their evolution of species that interact with aerosols and determine the formation and evolution of the aerosol size distribution.

The mono-wavelength allows a very simple and computationally cheap implementation. In this project, the optical extinction at 1 μm is set as proportional to the aerosol density at a fixed mean particle radius. No production and destruction of these particles is considered, as imposed by a mono-wavelength approach. This simple mono-wavelength approach allows presenting the assimilation output immediately in optical extinction coefficients, and, providing a straightforward comparison between model and observations.

4.2.2 Spatial and temporal coverage of remote sensing data.

Of fundamental importance for the assimilation analysis is the spatial and temporal coverage of the remote sensing data. This coverage determines the domain of the assimilation analysis independently of the quality of the model. In this project, we have relied on a worst possible case with respect to spatial and temporal coverage, namely SAGE-II data.

Means to address the limitations by these two considerations will be extensively discussed later.

4.2.3 Description of the 4D-Var assimilation model.

The 4D-Var assimilation is build on three major components:

1. The model that describes the evolution of the stratospheric aerosols forward in time (= model),
2. The adjoint of the model, when calculated backwards in time gives the gradient of the discrepancy of model values and observations with respect to the model initial conditions (= adjoint model).
3. The optimisation algorithm, tuned for large-scale problems, finds model initial conditions which minimise the discrepancy between model and observed values (= optimisation algorithm).

4.2.3.1 Model description.

The model is a 3-D transport model with a semi-lagrangian advection scheme. The horizontal resolution is variable: a T21 spectral resolution truncation, $\approx 5.6^\circ \times 5.6^\circ$, (default), a T42 truncation, $\approx 2.5^\circ \times 2.5^\circ$, and, a T63 truncation, $\approx 1.8^\circ \times 1.8^\circ$. In the vertical, 5 stratospheric levels are possible, 10, 30, 50, 70, and, 100 mbar. Only the lowest resolution is considered in the current version. Neither microphysical processes nor extra stratospheric sources/losses are included in this version of the model. The model time-step is by default is 6 hours, but can range from 30 minutes up to 24 hours by multiples of 30 minutes. For each time-step t_i , observations taken in the period $t_i - \Delta t/2$, $t_i + \Delta t/2$ are considered as simultaneous. Due to the spatial coverage of the SAGE-II data, spatial mapping of the observations was done onto the nearest model grid point.

4.2.3.2 Adjoint model.

The adjoint model is created by automatic differentiation of the model. This implies for the code of adjoint model: easy maintenance, reliability, optimisation issues take into account, and, standard programming language.

4.2.3.3 Optimisation algorithm.

The optimisation is based on the variable metric quasi-Newton method. Currently the package of MINPACK-II is used for medium-sized problems. These routines are in the public domain: <http://www.mcs.anl.gov/summaries/minpack93/summary.html>. A second method in the same family, believed to be faster for large-sized problems, will be implemented shortly: M1QN3. These routines were developed especially by INRIA (France) for meteorological applications (ECMWF, Météo France) and are only available on special request.

These methods are iterative algorithms and require for each iteration at least one evaluation of the forward model and one evaluation of the adjoint model. In the computational cost breakdown, the optimisation algorithm takes about $\approx 1\%$, the rest is taken by the model and its adjoint. This implies that the number of function and gradient evaluations must be kept as low as possible.

The implementation of the optimisation algorithm in the assimilation process requires user input for the convergence criteria (default settings are available):

- Number of forward model evaluations (100),
- Number of backward adjoint model evaluations (100),
- Relative tolerances on “cost” decrease (10^{-9}).

With the use of the MINPACK-II routines, several difficulties arise mainly due to the goodness of the first guess. Therefore two possibilities in solving the optimisation problem: use $\ln \mathbf{x}_0$ and $\ln J$ as the independent and dependent variable or \mathbf{x}_0 and J . Current experience has learned that for a *bad* first guess the use of $\ln \mathbf{x}_0$, $\ln J$ is more appropriate, and for a good first guess \mathbf{x}_0 , J is more suitable. The quantification of a good or bad initial guess is still not yet defined. A general solution (default) is to implement the first iterations with $\ln \mathbf{x}_0$ to solve and later on switch to \mathbf{x}_0 . Final decisions will depend on the results of the M1QN3 algorithm. This comparison is crucial for the further improvements on the assimilation analysis system.

4.2.4 The first guess of the model initial conditions.

Since in this project the assimilation analysis is not part of an operational analysis chain, special attention has to be given to the preparation of the first guess of the model initial conditions. For the SAGE-II data the following method has been developed for a given assimilation period:

For each observation a back-trajectory until assimilation start time is calculated. The concentrations corresponding to the starting point of the back-trajectory are then used in the first guess. This technique is called back-trajectory mapping. In this implementation, a possible divergence of the back-trajectories is not verified. The gaps in the model distribution are filled with a zonal mean calculated from the observations for the given assimilation period and extrapolated towards the poles. Currently this technique gives the best results. In this case lnJ is minimised.

The output from a previous assimilation process can also be used as first guess of the model initial conditions. This is done to find the best possible solution of the optimisation algorithm and is performed as a second phase in the assimilation process.

4.2.5 Aerosol assimilation flowchart and code description

```

INPUT:  Start and end date of assimilation period
        Spatial and temporal (= model time step) resolution (currently on T21 horizontal resolution)
        RUN identification (4 characters)
        Maximum number of model and adjoint model evaluations

ASSIMILATION SET-UP:  Model set-up: resolution, number of time steps
                       First guess for model initial conditions:
                           Zonal mean of available observations +
                           Trajectory mapping for assimilation period (first phase)
                           Output of previous assimilation analysis (second phase)
                       Set-up for  $\partial lnJ/\partial ln x_0$  (first phase) or  $\partial J/\partial x_0$  (second phase)

OPTIMISATION ITERATIONS
  Do for ITER= 1, ITMAX

    MODEL EXECUTION ( $x_m = M_{m-1} x_{m-1}$  Eq. 18)
      Do time-step = 1, final time-step
        Read dynamical input (t, u,v,w, psurface)
        Read observations for corresponding time
        Advect species
        Update cost functional
        Write species distribution, and additional info for adjoint
        Calculations
        Write model species and observations for analyses
        Write cost functional J, total and for each model level
      Enddo
    OUTPUT: cost functional J

    ADJOINT MODEL EXECUTION (Eq. 22)
      Do time-step = final time-step, 1, -1
        Read species and necessary info for adjoint calculations.
      Enddo
    OUTPUT:  $\partial lnJ/\partial ln x_0$  or  $\partial J/\partial x_0$ 
    WRITE:  $\partial lnJ/\partial ln x_0$  or  $\partial J/\partial x_0$  for further analyses

    MINIMISATION ROUTINE
      New initial model conditions for MODEL
      Test reduction J / convergence criteria if convergence goto end convergence

  Enddo
  End convergence
  OUTPUT: INITIAL CONDITIONS for MODEL
END.

```

ASSIMILATION EXECUTION: BATCH UNIX SCRIPT

The assimilation process execution is currently done in a BATCH environment on a CRAY-J90. Since there is not much input required from the user, this necessary input (= run control variables) are passed to the program by the FORTRAN namelist mechanism. The following example of a BATCH UNIX script shows 2 phases into the assimilation process. The first phase uses a first guess of the model initial conditions derived from the available remote sensing data, The second phase uses the result from the first phase of the assimilation process as a first guess. In the first phase lnJ is minimised and in the second phase J is minimised. This is due to the quality of the respective first guesses.

```
#QSUB batch parameters
#QSUB
SRC_DIR=/home/4dvar/MASTER/v1      # set source code directory
OUT_DIR=/tmp/4dvar                  # set output directory

cd $SRC_DIR
ja                                  # set job accounting on
#BEGIN ASSIMILATION
#compile source code for first phase
cf77 driver_lnx_ini.f model_lnx_const.f admodel_lnx.f dvmlm.f -o t21_lnx_ini

#execution of first phase
# kabnum is the total number of time-steps for the model
# delt is the time-step in seconds
# itmax is the maximum number of function and gradient evaluations
# file_ext is an identifier of the assimilation process by which the output of the assimilation process can #
studied
# ini_day is the initial date of the assimilation process (for development purposes only this is the day of #
January 1989
t21_lnx << EOF
&INPUT
  kabnum=36,
  delt=7200,
  itmax=200,
  file_ext='dC33',
  ini_day=10
&
&END
EOF
#End of first phase
# compile source code for second phase
cf77 driver_x.f model_x_const.f admodel.f dvmlm.f -o t21_x
#use output from first phase as first guess by copying into standard file con0
cp $OUT_DIR/100000_c.dC33 $OUT_DIR/con0
t21_x << EOF
&INPUT
  kabnum=36,
  delt=7200,
  itmax=500,
  file_ext='dC34',
  ini_day=10
&
&END
EOF
#put accounting off
ja -cst
#END ASSIMILATION
```

4.2.6 Output and file formats

After the execution of the assimilation process not only the solution (= model initial conditions), but also the distribution of the assimilated fields is available at the imposed model resolution for the assimilation period. For analysis purposes, the model assimilated fields, model temperature, observations and observation errors are outputted. To save disk space, per time-step the output is written to 2 files with the following conventions: "out_dir/fnum/'_?'/'file_ext":

Out_dir = specified output directory,

Fnum = DDHHMM of January 1989 (currently) D = day; H = hour; M = minute

_? = _c for global distribution of assimilated fields

_? = _o for longitudes, latitudes, pressures, assimilated fields, model temperature, observations and observation errors, at locations where there were observations taken into account in the assimilation process.

File_ext = four character run identifier.

The output format is given in the following code of output subroutine:

```
open (2,file=out_dir/fnum/'_c'/'file_ext)
open (15,file=out_dir/fnum/'_o'/'file_ext)
  do k=1,nlong          ! nlong = number of model longitudes
  do i=1,iimax         ! iimax = number of model latitudes
  do j=1,jjmax         ! jjmax = number of model stratospheric levels
C conc = array containing model fields to be assimilated
C con_obs = observations
C con_error = observation errors
C xlev = model longitudes
C ylev = model latitudes
C zlev = model pressures
C t = model temperatures
    write(2,902)conc(k,i,j)
      if (con_obs(k,i,j).le.999.0)      ! check for observations presence
$   write(15,903)xlev(k),ylev(i),zlev(k,i,j),conc(k,i,j)
$   ,con_obs(k,i,j),con_error(k,i,j),t(k,i,j)
    enddo
  enddo
  enddo
close(2)
close(15)
1 continue
902 format(e13.4)
903 format(2(f6.2,2x),5(e13.4,2x))
```

Also the evolution of the cost functional J is available for analysis.

4.2.7 IDL assimilation visualisation and analysis tool.

A user-friendly interface has been developed to visualise and analyse the results of the assimilation process. An IDL widget based interface was built to view the outputs of the assimilation process. Our main requirement were:

- Contour maps of analysed species for the complete assimilation process at each model time-step, each model level, and, with zooming capabilities.
- The evolution of the cost functional as function of the optimisation iteration.
- Vertical profiles of coincident model values and observations.
- Comparison of different neighbouring model vertical profiles.
- Printing of graphical results.
- Easy access to output from different runs.

As an example, the appearance of the main interface window is given in figure wp1200.2. It contains three menus ('File', 'Color' and 'Plot'), four drop lists ('level', 'soomthparam.', 'period' and 'run'), one list

(with the elements ‘contour’, ‘cost’ and the location where we could find SAGE-II data for that period) and three buttons. The ‘File’ menu has all the buttons, which lead to open a file, print a plot, exit the program. The ‘Color’ menu allows the user to select color and the ‘Plot’ menu to change the plot characteristics.

By changing the elements of a drop list, the interface will update the graphical result for the newly requested situation. The drop list ‘level’ controls the altitude of the contour map. All the altitude levels of the assimilation are available (10, 30, 50, 70 and 100 mbar). The ‘period’ and ‘run’ drop list control the time period and the run of the analysed field to be loaded. By clicking on the different elements of the list we move from the actual view to the requested one. By clicking on ‘refresh’, the program reloads the same data file. This is very useful when we want to see the evolution of an assimilated field during a run. The last two buttons ‘Zoom’ and ‘Select profiles’ are exclusive. They control the action which follows when the user has selected an area on a contour map with the mouse. If ‘Zoom’ is set, a second window is created with a zoom for this area. Otherwise, if ‘Selected profiles’ was chosen, a third window appears with all the vertical profiles of the extinction coefficient for this area.

4.3 Validation of 4D-Var assimilation framework.

Two simple but conclusive experiments were set up for testing the assimilation framework:

- The model was run for a number of time-steps (60×6 hrs) starting with some imposed initial conditions \mathbf{x}_{test} . All these model values were then used as observations in an assimilation procedure for 2 time-steps with a uniform error of 1 %. In this procedure the initial guess of the assimilation was chosen $\mathbf{x}_0 = 2 \times \mathbf{x}_{\text{test}}$. This implies that $2 \times 64 \times 32 \times 5 = 20420$ observations were used in the assimilation procedure, and, after 150 iterations a solution or an assimilation analysis was found within 1% of \mathbf{x}_{test} . This test proves the validity of the overall assimilation procedure.
- The same model calculated observations were used in an experiment to study the behaviour of the assimilation framework used with a low spatial and temporal coverage of data. Per time-step and for each level observations were taken at 1 latitude and 2 longitudes 180° apart starting at highest latitude (North Pole) and 0° and 180° longitude, the following time-step, the observation was taken 1 latitude grid point south and 1 longitude grid point east. The same procedure was repeated for each time-step. The assimilation time window was 10 days. The first guess for the assimilation was taken $\mathbf{x}_0 = 1.2 \times \mathbf{x}_{\text{test}}$. The spatial domain of \mathbf{x}_0 that is influenced by the assimilation can be inferred by the gradient of the cost functional J with respect to the initial conditions. The higher values of the gradient are a signature of the evolution of the artificial observations (Figure wp1200.3). The back-tracing of the influence is clear from the “diffusive character” of the gradient due to the advection or transport by the winds in the stratosphere. A solution within 5 % of the assimilation was found after 40 iterations. When the model is run with these initial conditions, the observations are reproduced within 1%.

4.4 Default settings for assimilation analysis.

The current default settings are based on the current experience. These settings will be optimised when the MQ1N3 algorithm is implemented (faster convergence), and, the assimilation is extensively used (more case studies) and a control mechanism is put in the system:

- The forward model time step: 6 hours (computational arguments).
- The assimilation window is limited to 10 days, because back-trajectories lose their significance beyond 10 day back-trajectories, longer periods are possible but require much more iterations of more computationally expensive model and gradient evaluations.
- Maximum number of forward model and backward adjoint model evaluations: 100 (computational arguments).
- Two phase assimilation process and first guess of initial model conditions accordingly.

4.5 Domain of validity of the assimilation analysis.

4.5.1 Spatial validity of the assimilation analysis.

The spatial validity of the assimilation analysis (= initial conditions for the forward model over the assimilation time window) will be determined by the coverage of the satellite and the winds in the stratosphere which back-trace the influence. If the first guess model initial condition is not the solution,

the gradient of the cost with respect to the first guess indicates the domain that will be covered by the assimilation. In figure wp1200.4 the gradient of the cost functional is shown for a five six-hour time step run starting at January 20, 1989. Available observations during this time window are concentrated along 2 longitudes well latitudinally separate. This shows that the satellite coverage plays a crucial role in the spatial domain in which the analysed fields are influenced by the observations. In this example, there is no exchange taking place between the Northern Hemisphere, equatorial region, and, Southern Hemisphere. This is completely what can be expected from the dynamical behaviour of the stratosphere. The valid assimilation domain could be identified as the domain where the gradient of the cost with respect to the initial conditions has decreased sufficiently. However, nor the quality or goodness of the first guess of the model initial conditions nor the observation errors are uniformly distributed. This means that the gradient information is only indicative of the spatial validity of the analysis.

4.5.2 Temporal validity of the assimilation analysis.

The physical processes included in the model description of the aerosols mainly determine the temporal validity of the assimilation window. Since only the simplest possible description of aerosols is allowed in the mono-wavelength approach, the temporal validity, in the case of the SAGE-II data will not allow a global assimilation analysis due to the poor data coverage. The temporal validity is currently limited to 5 days and even better results are obtained with a smaller time window. This is based on current experience and a better quantification of the temporal limitation will be the result of more extensive usage of the assimilation system.

4.6 Results.

For the discussion, results from two assimilation processes will be shown:

- A one-day assimilation analysis starting on January 10, 1989 (run identifier = dS12), and,
- a five-day assimilation analysis starting on January 10, 1989 (run identifier = dS54).

Both runs were performed with the default settings except for the model time-step, which was set at 2 hours. The was only done to obtain a better temporal output resolution, the 6 hour analysis does not differ fundamanently from the results presented here. Since the one-day analysis is based on a sub-set of observations of the five-day analysis, this allows a straightforward study of the influence of temporal extend of the analysis. The location of the observations used in these analyses is shown in figure wp1200.5. The coverage of the observations indicates that mainly due to the dynamical properties of the stratosphere, the analysis will be usefull in the northern sub-tropical and tropical region.

Figure wp1200.6 shows a comparison of analysed and observed vertical extinction profiles for both runs at January 11 at different locations. The excellent agreement of the analysis with observations is remarkable. The horizontal distribution of the analysed extinction is shown in figure wp1200.7 for both run for January 11, at 50 mbar. Both runs show the same general features, although the five-day analysis exhibits a higher spatial variability that is due to the higher information content in the analysis.

It is known that in tropical regions sub-visible cirrus clouds can be detected from the SAGE-II data. If sufficient observations are taken into account, this would lead to discrepancies between analysed field, using only this simple mono-wavelength approach, and observations at lower altitudes. Figure wp1200.8 shows for January 10 at 8 hour UT the vertical extinction profiles of the analysis and observations. For this time at lower model altitudes, a discrepancy between the five-day analysis and observations is apparent in the tropics. In the one-day analysis and at higher latitudes for both analyses the model agrees very well with the observations. The current explanation is that a sub visible cirrus cloud is present in the observations. For the one-day analysis not enough measurements are available for the model to make a distinction between aerosols and the sub-visible clouds.

Figure wp1200.9 shows the influence of a sub-visible cloud event and the influence of the spatial variability for both analyses on the vertical profile for neighbouring model grid points.

From these results it is clear that an analysis obtained by an assimilation process is able to reproduce the observations taking into account the limitations of the model and the limitations due to the spatio-temporal coverage of the remote sensing data. The analysis offers the possibility to address aerosol optical extinction information where no observations are available. Even with a simple aerosol model, we have shown that the analysis contains very usefull information about non-aerosol events in the optical extinction data and that this analysis can also be used to identify Polar Stratospheric Cloud events in the aerosol remote sensing data.

4.7 Future improvements.

Future improvements to mono-wavelength approach are expected from:

- The implementation of the M1QN3 algorithms (June 1998),
- More case studies and other remote sensing data. It is clear that due to the high variability of the real stratosphere, a large number of case studies are necessary to verify extensively the assimilation process. This is necessary for further improvements.
- Better model description of the stratospheric dynamical state. At this point, the model uses the dynamical fields of the stratosphere, which are based on a high time resolution analysis with the Numerical Weather Prediction model of ECMWF. This model will have a greatly improved description of the stratosphere, 20 stratospheric vertical levels up to 0.1 mbar, operationally by the summer of 1998.

Further improvements are expected from the multi-wavelength approach. This will allow a better physical model of the stratospheric aerosols, therefore leading to a better assimilation system.

4.8 Conclusions.

In this first attempt of 4D-Var assimilation of stratospheric aerosols, it has been shown that this technique offers great possibilities. However, a progress in the assimilation process can only be obtained by ongoing further improvements based on experience learned from a great number of case studies.

Nevertheless, the 4D-Var assimilation process has already proven to be the tool to address local spatial and temporal information of stratospheric aerosols, based on nearby remote sensing data as been shown by the results.

The presented assimilation framework has the great advantage of flexible and easy extension of the model to include a multi-wavelength approach with a microphysical description of the aerosols, and, to include assimilation of chemical species including their chemical transformations.

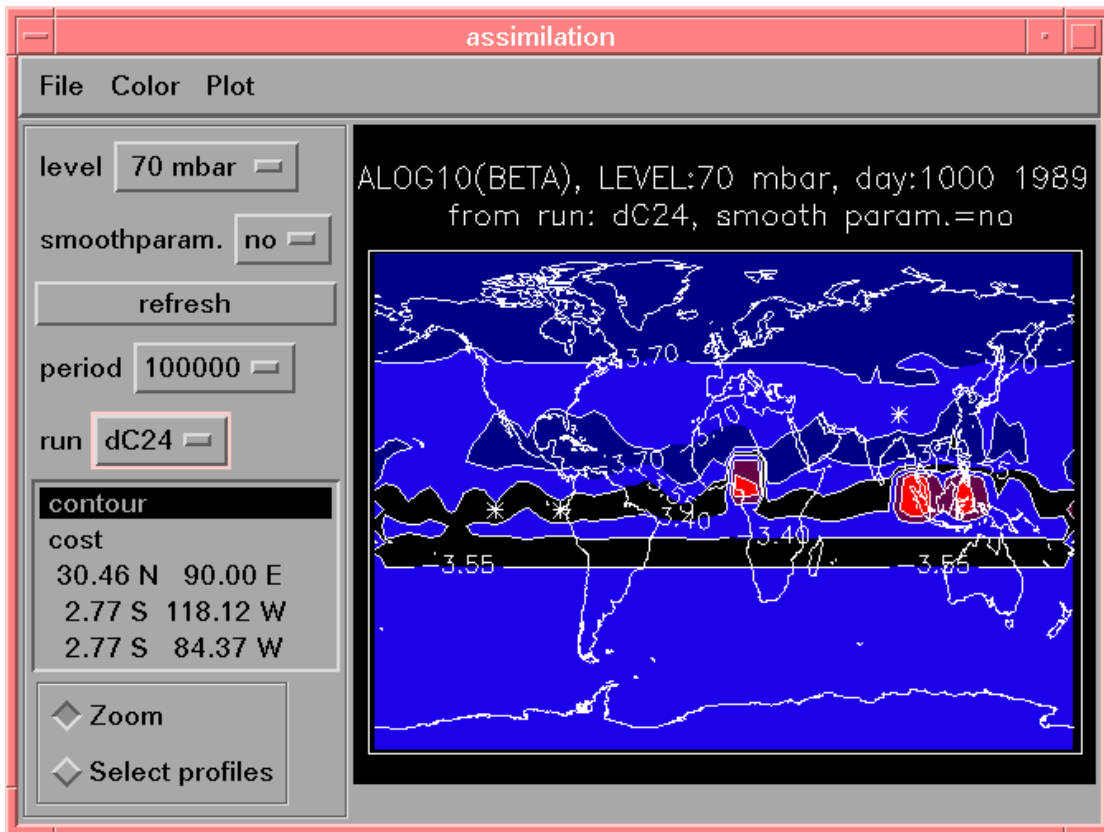


Figure wp1200.2: Main interface window of visualisation and analysis tool.

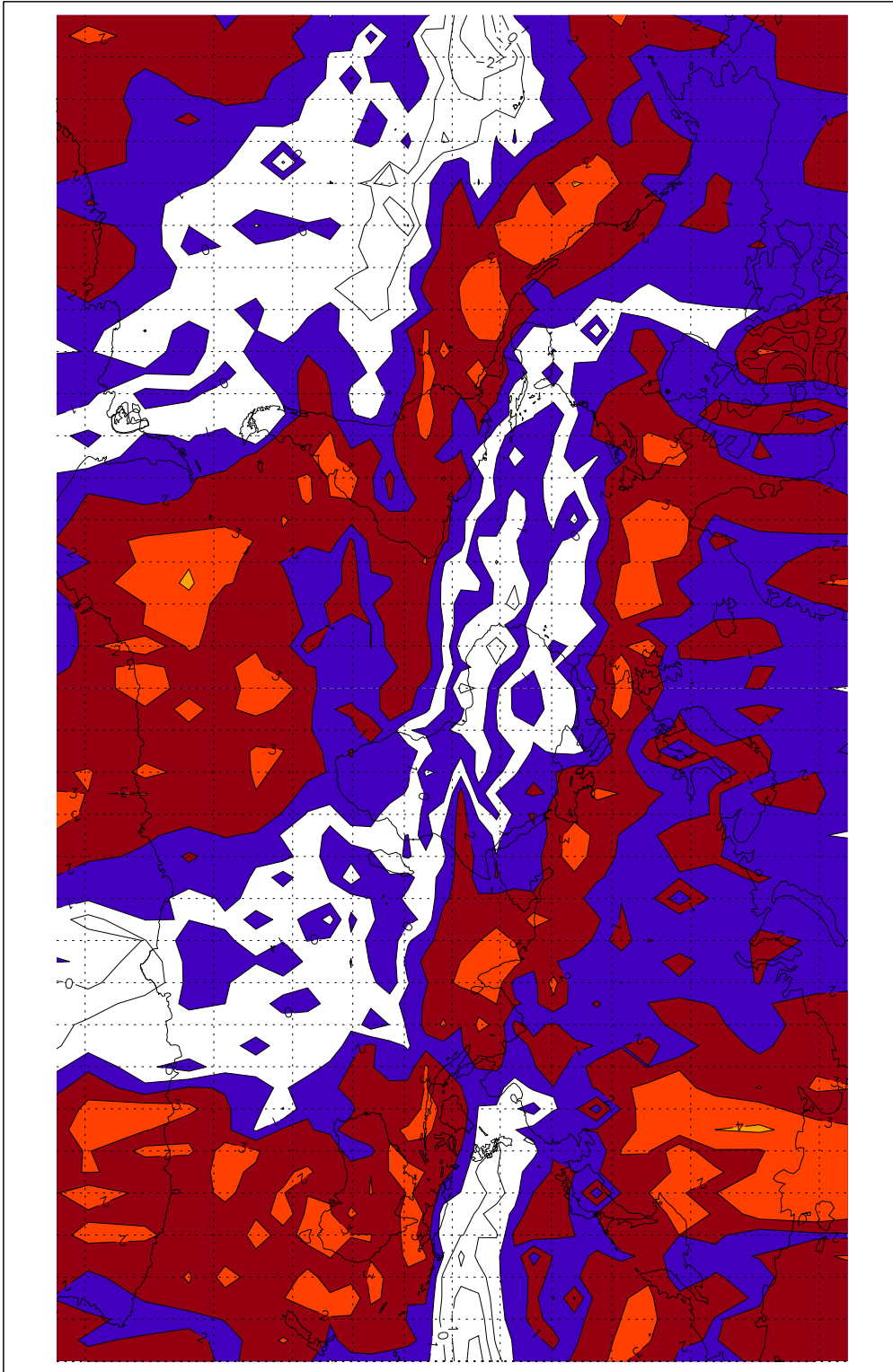
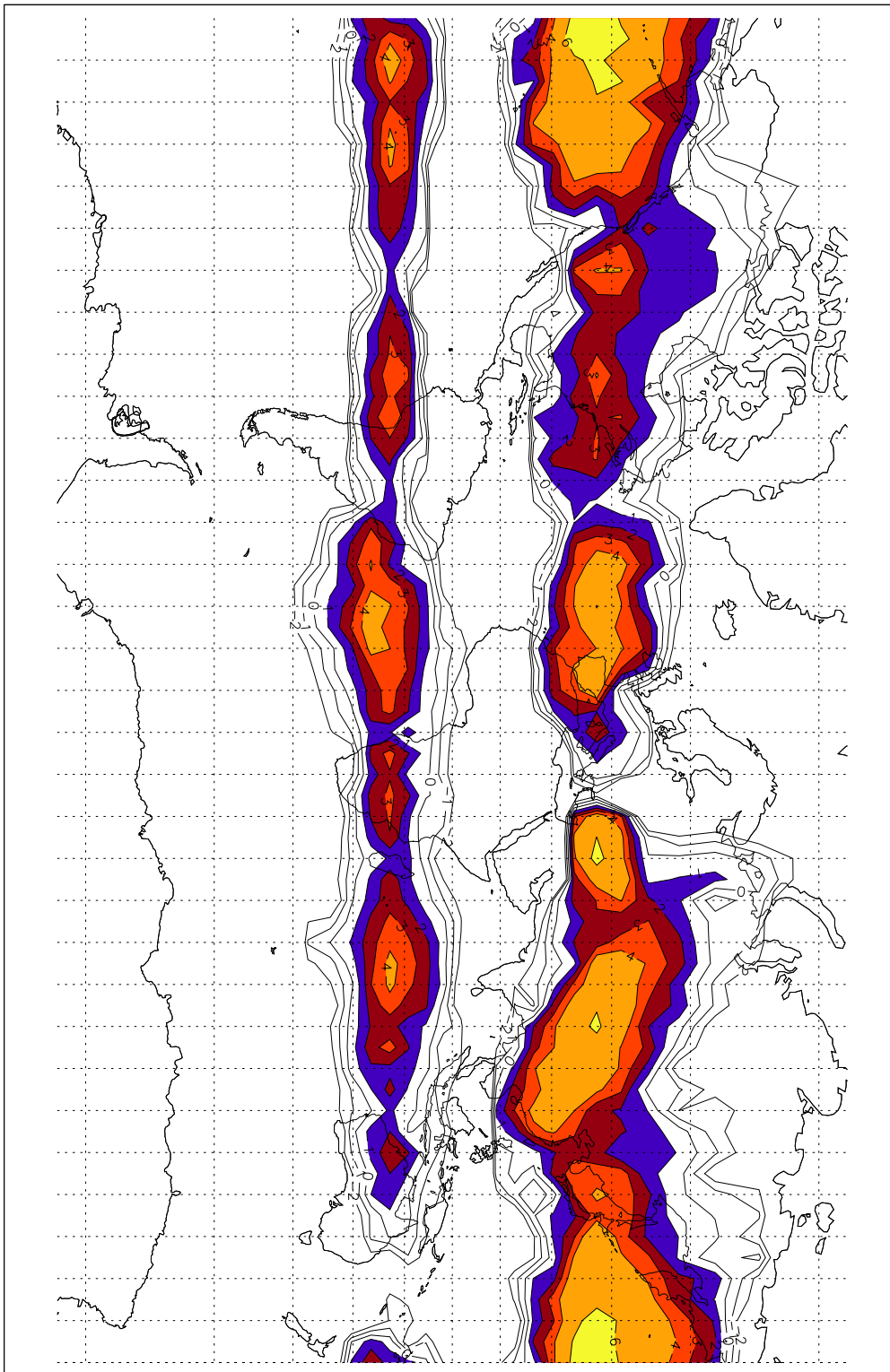


Figure wp1200.3: Gradient of cost functional J with respect to model initial conditions for second test case, contour levels are given in powers of ten.



INITIAL COST GRADIENT for RUN = d241

Figure wp1200.4: The gradient of the cost functional J with respect to the initial model conditions, values are in powers of ten (See text for explanations).

Location of the SAGE-II observations
for the runs dS12 (triangles) and dS54 (triangles & squares)

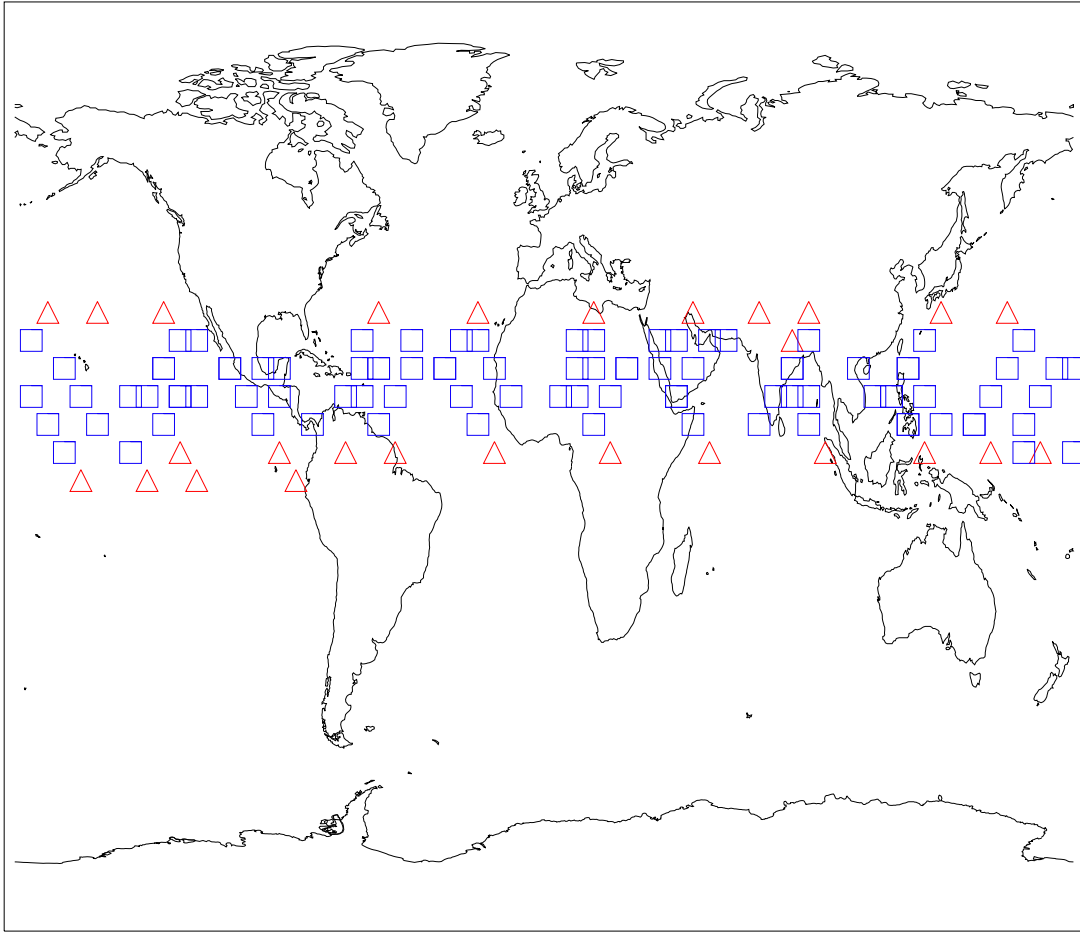


Figure wp1200.5: Locations of the SAGE-II observations for analyses starting on January 10 for a one-day period (squares) and a five-day period (squares and triangles).

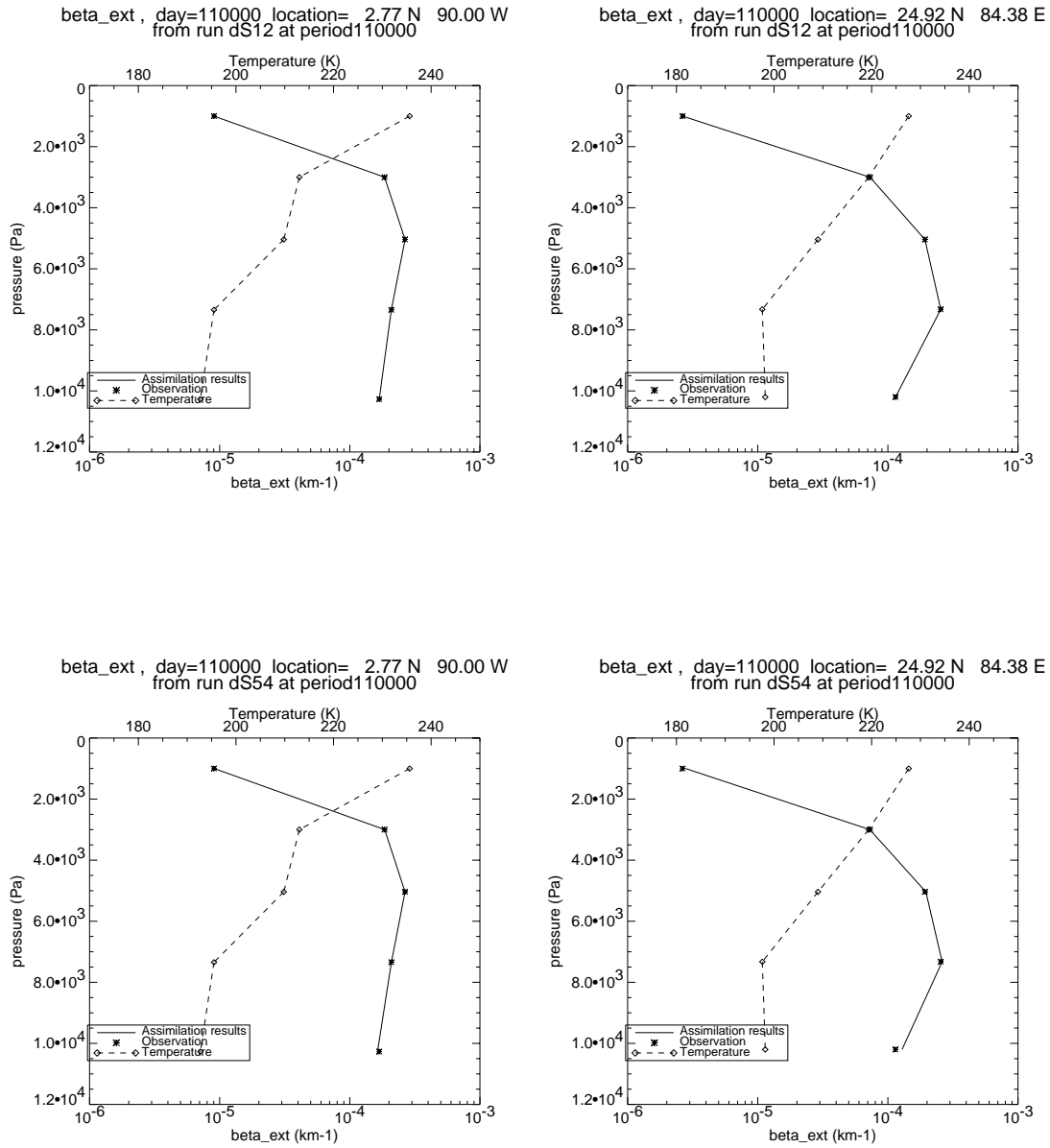
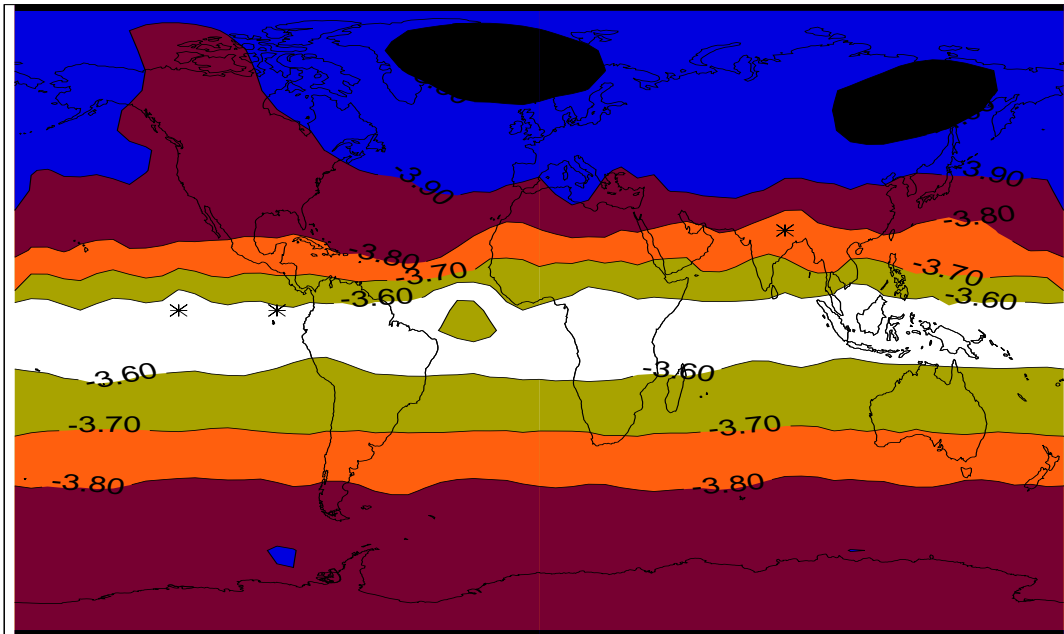


Figure wp1200.6: Analysed vertical extinction profiles and observations.

ALOG10(BETA), LEVEL:50 mbar, day:1100 1989, from run: dS12
smooth param.=no



ALOG10(BETA), LEVEL:50 mbar, day:1100 1989, from run: dS54
smooth param.=no

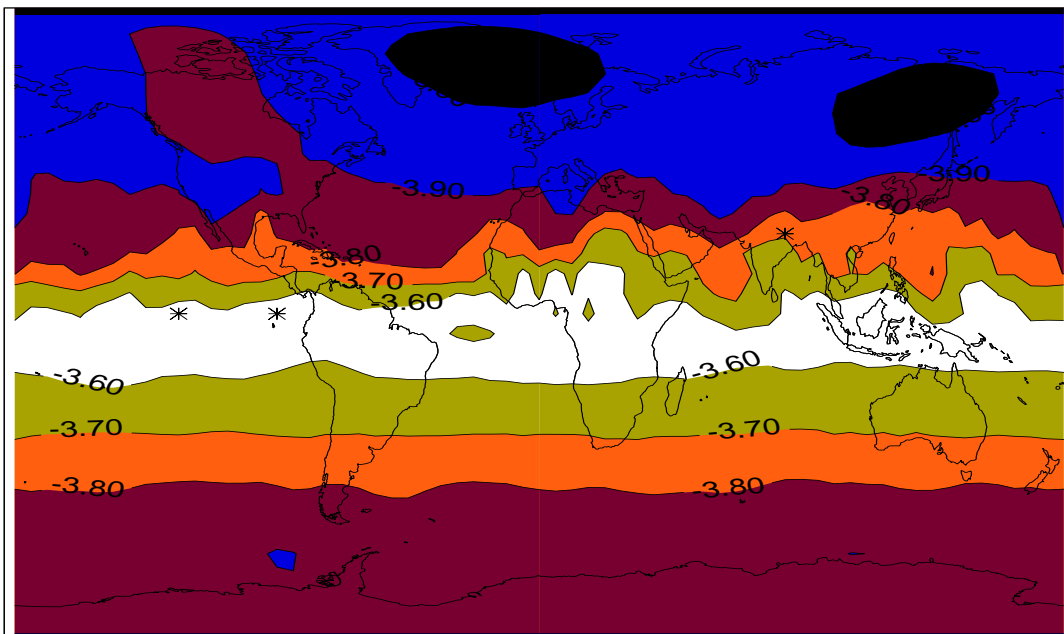


Figure wp1200.7: Global distribution of analysed optical extinction at 50 mbar for January 11 for a one-day and a five-day analysis.

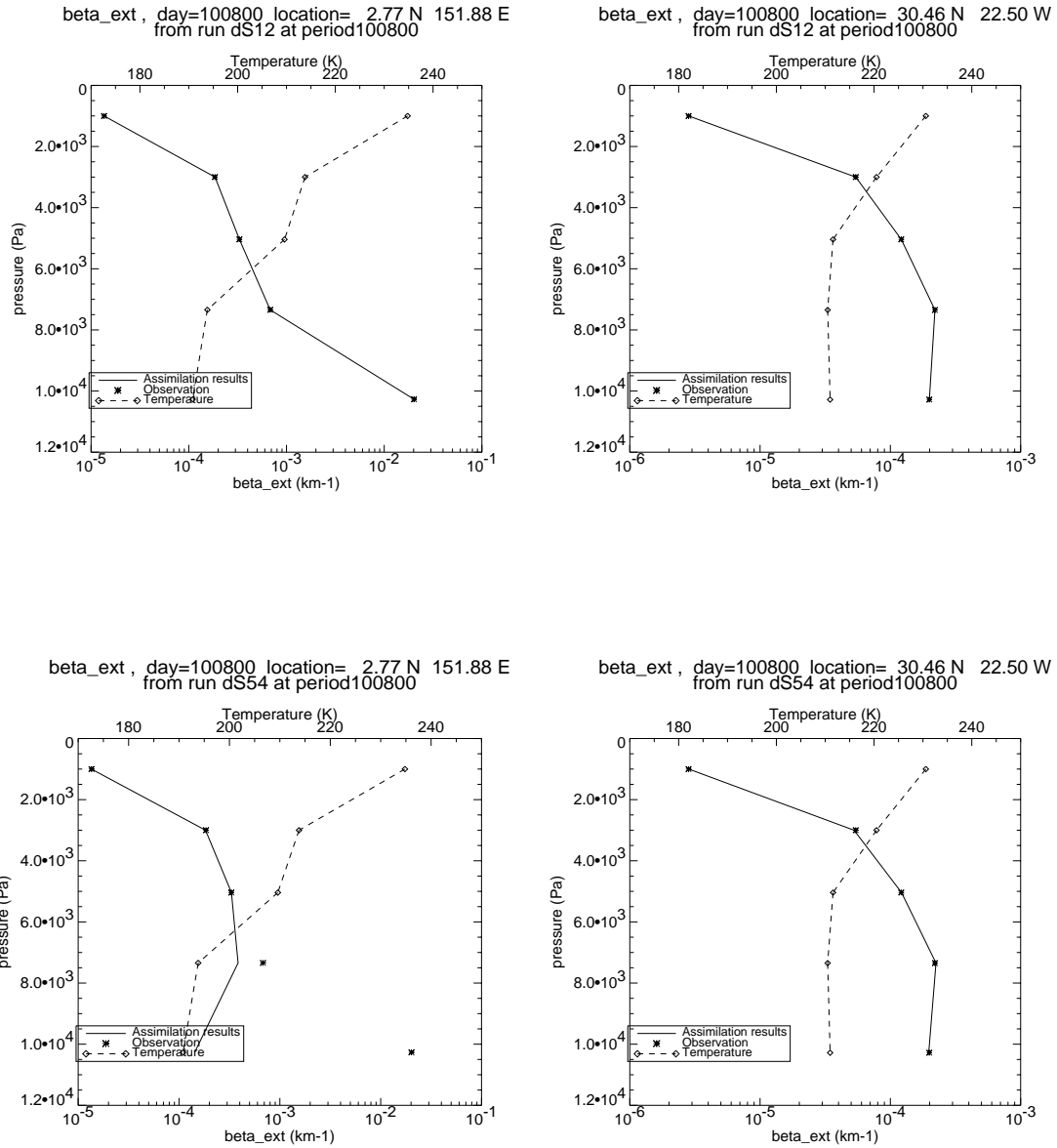


Figure wp1200.8: Analysed vertical extention profiles and observations.

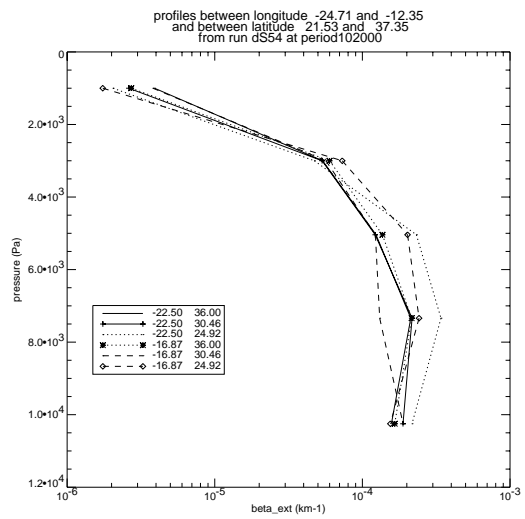
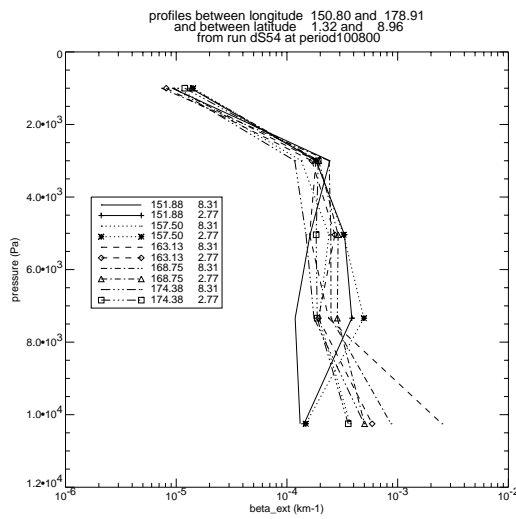
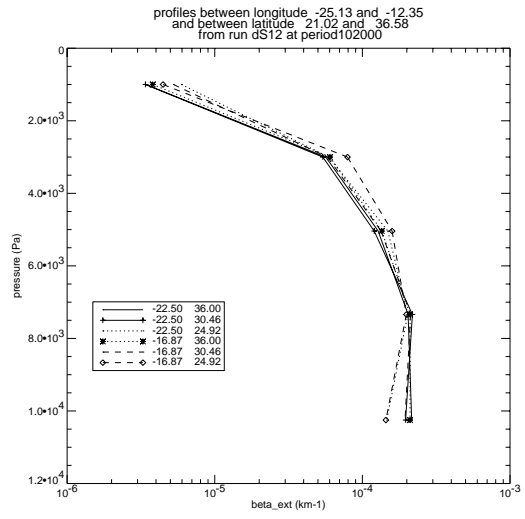
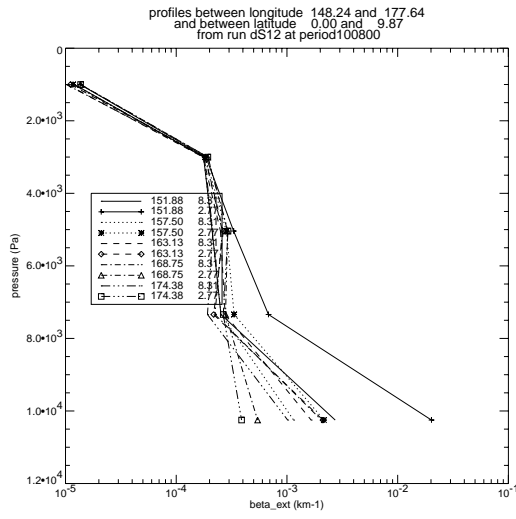


Figure wp1200.9: Analysed vertical extention profiles and observations.

5 Migration toward future ENVISAT data

At present, it looks like the mapping tools can be adapted straightforwardly to the future ENVISAT missions, under the same conditions (i.e., adopting the same limitations as the ones that have been set actually):

The layered structure of the cartography software (section 3.1) and the conceptual approach of the assimilation tool make it easy to integrate different data sets (different experiments, different species). The implementation of the aerosol assimilation is already prepared for the inclusion of a more detailed description of aerosols and of other chemical species in the assimilation process.

Moreover, ENVISAT should obtain global coverage in a much shorter time than SAGE-II, which means that the temporal resolution of the results will be higher. In particular, ENVISAT should cover nearly all latitudes in one orbit which is a far better situation than the one of SAGE-II (2 latitudes, and 16 longitudes per latitude in one orbit) as far as aerosol assimilation is concerned. Indeed, the stratospheric transport being essentially in the zonal direction, data assimilation allows to ‘complete’ the assimilated map mainly along latitude circles, far better than along the meridians.

6 Conclusions

Mapping tools for satellite level-2 aerosol data have been developed successfully in the frame of the DAMS2P project:

- Algorithms are available for mapping based on interpolation on the one hand (cartography), and through 4D variational data assimilation on the other hand, which is based on an atmospheric transport model with real (ECMWF) winds in which the aerosols are considered to be purely dynamical tracers. Such algorithms that come up to the specific problems associated with the satellite level-2 data were not available earlier.
- These algorithms have been implemented and tested, and they have been validated to a limited extent. The effects of data uncertainties have been evaluated.
- Software interfaces and data formats are clearly defined. A number of user parameters have been identified, which allow some fine-tuning of the results to specific user needs.
- The production of aerosol maps can be performed on the local platforms; example maps are available and can be produced upon request.

The above developments and verifications are based mainly on the SAGE-II stratospheric aerosol extinction data at a single wavelength. The extinction data have been converted to particle densities through a most simple aerosol model that assumes that the number density is proportional to the extinction.

The cartography algorithm can be applied to tropospheric aerosol as well. However tests with preliminary GOME total extinction data indicate that there are still problems with these data that have to be discussed. The assimilation tool is limited to stratospheric aerosol because of the atmospheric model behind it.

Application of the algorithms to data from future satellite missions like ENVISAT should be straightforward.

The perspectives are:

- in the near future:
 - ⇒ more extensive validation of the algorithms, among others through mutual intercomparison
 - ⇒ integration of multiple data sets in one map, on the purpose of obtaining global coverage in a smaller time window.
 - ⇒ exploitation of multi-wavelengths aerosol data sets, on the purpose of deriving various aerosol properties and associated maps
 - ⇒ extension of the mapping to additional species, in particular H₂O and HNO₃
- possibly in a far future:
 - ⇒ user-oriented interfaces, possibly via WWW, for exploitation by a larger user community

Presentations of the present work at scientific workshops prove that there exists a real need for such tools. They enable a better visualisation of the data and easy the interpretation. In particular the data assimilation will serve validation purposes of satellite atmospheric chemistry data. The future ENVISAT mission will be an interesting opportunity for exploiting the mapping tools that are being developed here.

7 Bibliography

- A. Ahlin, "A bivariate generalisation of Hermite's interpolation formula", *Mathematics of Computation*, vol. 18, pp. 264-273, 1964.
- Bischof C. et al., ADIFOR 2.0 User's Guide, Mathematics and Computer Science Division, Technical Memorandum N° 192, Argonne National Laboratory, 1995
- P. Ciarlet and P. Raviart, "General Lagrange and Hermite interpolation in R^N with applications to finite element methods", *Arch. Rational Mech. Anal.*, vol. 46, pp. 177-199, 1972.
- P. Davis, "Interpolation and Approximation", Dover, New York, 1975.
- P. Dierckx, "Curve and surface fitting with splines", Clarendon Press, Oxford, 1995.
- M. Fisher and D. J. Lary, Lagrangian four-dimensional variational data assimilation of chemical species, 121, 1681-1704, 1995.
- G. Franssens, "A new non-polynomial univariate interpolation formula of Hermite type", in preparation, 1998.
- M. Gasca and J. Maetzu, "On Lagrange and Hermite Interpolation in R^N ", *Numerische Mathematik*, vol. 39, pp. 1-14, 1982.
- R. Giering, Tangent linear and Adjoint Model Compiler: User manual, Center for Global Change Sciences, MIT, Cambridge, USA, 1997
- M. Mochi, et. al., "GOME data processing at I-PAF: the aerosol optical thickness retrieval from GOME spectra", to appear in the Proceedings of the 3rd ERS Symposium 'Space at the Service of our Environment', Florence, 17-21/03/1997a.
- M. Mochi, "Description of the test aerosol GOME product", Technical Note No. S792-ITD-NT-001, 15/10/1997b, Nuovo Telespazio s.p.a.
- R. Renka and A. Cline, "A triangle based C^1 interpolation method", *Rocky Mountain Journal of Mathematics*, vol. 14, pp. 223-237, 1984.
- R. Renka, "Interpolation of data on the surface of a sphere", Oak Ridge National Laboratory, report ORNLICSD-108, 1982.
- R. Renka, "Algorithm 623", *ACM Transactions on Mathematical Software*, vol. 10, p. 437, 1984.
- Shepard, "A two-dimensional interpolation function for irregularly spaced data", *Proceedings of the 23th National Conference of the ACM*, Brandon/Systems Press Inc., Princeton, USA, pp. 517-523, 1968.
- O. Talagrand and P. Courtier, Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory, *Q. J. R. Meteorol. Soc.* 113, 1311-1328, 1987.
- C. Trepte, et. al., "Banded structures in stratospheric aerosol distributions", *Geophysical Research Letters*, vol. 21, no. 22, pp. 2397-2400, 11/1994.

Appendix A. List of acronyms

ADEOS	Advanced Earth Observation Satellite
AMASDU	Aerosol Mapping Algorithms for Satellite Data Users
BIRA/IASB	Belgian Institute for Space Aeronomy
DUP	Data User Program
EC	European Commission
ENVISAT-1	Environmental Satellite-1
ERS-2	European Remote Sensing Satellite-2
ESA	European Space Agency
ESRIN	European Space Research Institute
EURECA	European Retrievable Carrier
GIF	Graphics Interchange Format
GOME	Global Ozone Monitoring Experiment
GOMOS	Global Ozone Monitoring by Occultation of Stars
HP	Hewlett-Packard
IDL	Interactive Data Language
IPCC	Intergovernmental Panel on Climate Change
JPEG	Joint Photographic Experts Group
MERIS	Medium Resolution Imaging Spectrometer
MT	Multipoint Taylor
ORA	Occultation Radiometer
PGADP	Prototypal GOME Aerosol Data Processor
POLDER	Polarization and Directionality of the Earth's Reflectances
PSC	polar stratospheric cloud
SAGE-II	Stratospheric Aerosol and Gas Experiment-II
SAM	Stratospheric Aerosol Measurement
TIFF	Tagged Image File Format
UARS	Upper Atmosphere Research Satellite
UNEP	United Nations Environment Program
WP	Work Package
WMO	World Meteorological Office

Appendix B. Appendix B: Cartography algorithm mathematical description

B.1 The interpolation problem

Aerosol observations done by satellites have up to now always used the limb scanning solar occultation method. This implies that measurements can only be done at sunrise and sunset satellite positions. Depending on the inclination of the satellite orbit (typical 60 degrees) and the number of revolutions, this results in a collection of measurement points which cover the larger part of the globe. It will be clear that this sampling of the atmosphere is highly irregular and is such that mid-latitudes are highly sampled, tropical regions less sampled (depending on the period of the year), while polar regions are uncovered. In order to ease the interpretation of the data it is necessary to interpolate the sample data to a regular and fine grid suitable for mapping purposes.

The GOMOS experiment on board the ENVISAT mission (to be launched in 1999) for the first time will use stellar occultation. This will highly improve the global coverage of the atmospheric sampling, but interpolation will still be necessary to produce maps. However the interpolation problem is expected to be easier and leading to more accurate results.

The interpolation problem has been considered from a general point of view and posed in full generality. Its mathematical formulation can be given as follows. Consider an ordered set of N distinct sample points on the unit sphere, $S = \{x_n = (\phi_n, \theta_n), 1 \leq n \leq N\} \subset S^2$ and an unknown real-analytic (i.e. C^∞) function $f : S^2 \rightarrow \mathbb{R}$, such that $x = (\phi, \theta) \mapsto f(\phi, \theta)$, with functional and derivative sample values $F = \{f_n^{k,l}, 0 \leq k+l \leq M_n, 1 \leq n \leq N\} \subset \mathbb{R}$ given at S , where

$$f_n^{k,l} = \left. \frac{d^{k+l} f(\phi, \theta)}{d\phi^k d\theta^l} \right|_{(\phi, \theta) = (\phi_n, \theta_n)} \quad (\text{B.1.1})$$

We want to find an interpolant $\tilde{f} : S^2 \rightarrow \mathbb{R}$, such that $x = (\phi, \theta) \mapsto \tilde{f}(\phi, \theta)$ and

$$\left. \frac{d^{k+l} \tilde{f}(\phi, \theta)}{d\phi^k d\theta^l} \right|_{(\phi, \theta) = (\phi_n, \theta_n)} = f_n^{k,l} \quad (\text{B.1.2})$$

This is the two-dimensional spherical version of the one-dimensional Hermite-Birkhoff interpolation problem on the real line. The Hermite interpolation problem is the generalisation of the Lagrange interpolation problem, where only functional values are prescribed. Various solutions have been given to the multivariate planar Hermite-Birkhoff problem [Ahlin, 1964; Ciarlet and Raviart, 1972; Gasca and Maetzu, 1982]. These solutions typically construct multivariate global polynomials through the data. These polynomials are often of high degree and this can lead to unwanted oscillations of the interpolant between sample points. In addition, for an interpolation method in two or more dimensions to be useful, its computational cost should be linear in time with the number of sample points. The number of practical two-dimensional interpolation schemes for scattered data is therefore rather limited.

Examples are the NAG routines E01SAF using the method of Renka and Cline [1984] and E01SEF using a modified Shepard method [Shepard, 1968]. The NAG library nor the Numerical Recipes package contain spherical interpolation algorithms. One of the rare spherical algorithms is ACM algorithm 623 from Renka (SPHPAK) [Renka, 1982; Renka, 1984]. This algorithm is implemented by IDL. It uses local cubic Hermite interpolation polynomials over geodesic arcs between two neighbour sample points and estimates local spatial gradients from a local quadratic approximation to the data. The interpolants over adjacent triangles are blended together by linear weighting. The resulting global interpolant is continuous up to first derivatives.

Below we give a general and more natural solution to the Hermite-Birkhoff interpolation problem. The proposed method is in theory applicable to any number of dimensions and has been developed for both planar as spherical interpolation. It has the following four major features:

- (i) Domain decomposition, using a Delaunay spherical triangulation;
- (ii) Interpolant representation is based on a truncated MT series;
- (iii) Local thin plate spline fit, used to estimate all spatial derivatives;
- (iv) Final filtering to remove short scale variations below the sampling limit.

B.2 Domain decomposition

For practical computational purposes, it is desirable to have a local interpolation algorithm. This means that the interpolated value at a point is only influenced by the data at the closest sample points. Then the computational cost of the algorithm becomes linear in the number of sample points. This is a necessary condition to be of practical use in two or more dimensions. The local nature of an interpolation algorithm is most conveniently introduced by introducing a decomposition of the interpolation domain.

A very convenient decomposition method is Delaunay triangulation. It has the property that within the circumscribed circle of a triangle, having three sample points as vertices, no other sample point is found. This will assure us that, the interpolation points inside a triangle, are only influenced by the closest sample points. Delaunay triangulation is very time efficient and works for any set of irregularly scattered sample points. Only when applied to a regular grid, this decomposition fails. Since in practice satellite measurements are never on a regular grid, this is no problem.

To make the interpolation local one needs basis functions with compact support. As support for our basis functions we use the interior of the Voronoi polygon around each sample point. The Voronoi polygon is defined as follows. Let x_n denote a sample point and the (anti-clockwise) ordered set

$X_n = \{x_i, i = 1, N_n\}$ its neighbours, such that

$$V_n = \{T_{i,n} = T(x_n, x_i, x_{i+1}), i = 1, N_n, x_{N_n+1} \equiv x_1\} \quad (\text{B.2.1})$$

are a set of Delaunay triangles $T_{i,n}$ having x_n as their common vertex. Then V_n is called the Voronoi polygon at the sample point x_n .

B.3 The Multipoint Taylor series representation

The new interpolation algorithm is based on a new mathematical concept to represent functions, called a Multipoint Taylor (MT) series representation. A MT series has the form

$$f(\phi, \theta) = \sum_{n=1}^N \sum_{0=k+l}^{+\infty} f_n^{k,l} U_n^{k,l}(\phi, \theta) \quad (\text{B.3.1})$$

wherein the $f_n^{k,l}$ are given by (B.1.1). For the $f_n^{k,l}$ to be the functional data at the sample points, the basis functions $U_n^{k,l}(\phi, \theta)$ must be $C^{\infty, \infty}$ and satisfy

$$\left. \frac{d^{i+j} U_n^{k,l}(\phi, \theta)}{d\phi^i d\theta^j} \right|_{(\phi, \theta) = (\phi_m, \theta_m)} = \delta_{mn} \delta^{ik} \delta^{jl}, \quad k+l \geq 0, \quad 1 \leq n, m \leq N \quad (\text{B.3.2})$$

Higher order basis functions $U_n^{k,l}(\phi, \theta)$, $k+l > 0$ can most conveniently be related to the zero order functions $U_n^{0,0}(\phi, \theta)$ by

$$U_n^{k,l}(\phi, \theta) = \frac{(\phi - \phi_n)^k}{k!} \frac{(\theta - \theta_n)^l}{l!} U_n^{0,0}(\phi, \theta) \quad (\text{B.3.3})$$

Conditions (B.3.2) then imply that the zero order basis functions $U_n^{0,0}(\phi, \theta)$ satisfy

$$\begin{aligned} U_n^{0,0}(\phi, \theta) \Big|_{(\phi, \theta) = (\phi_m, \theta_m)} &= \delta_{mn}, \quad 1 \leq n, m \leq N \\ \frac{d^{i+j} U_n^{0,0}(\phi, \theta)}{d\phi^i d\theta^j} \Big|_{(\phi, \theta) = (\phi_m, \theta_m)} &= 0, \quad i+j > 0, \quad 1 \leq n, m \leq N \end{aligned} \quad (\text{B.3.4})$$

Using (B.3.3) the MT series can be written as

$$\begin{aligned} f(\phi, \theta) &= \sum_{n=1}^N \left(\lim_{M \rightarrow +\infty} \sum_{0=k+l}^M f_n^{k,l} \frac{(\phi - \phi_n)^k}{k!} \frac{(\theta - \theta_n)^l}{l!} \right) U_n^{0,0}(\phi, \theta) \\ &= f(\phi, \theta) \left(\sum_{n=1}^N U_n^{0,0}(\phi, \theta) \right) \end{aligned} \quad (\text{B.3.5})$$

Herein it was assumed that the N ordinary Taylor series, evaluated around the N sample points for any point $x = (\phi, \theta) \in S^2$, converge to the same function $f(\phi, \theta)$. A sufficient condition for convergence of a MT series is therefore that $S^2 \subset \Omega_1 \cap \dots \cap \Omega_N$, with Ω_n the region of convergence of the n -th ordinary Taylor series. To ensure convergence to the correct function, the zero order functions $U_n^{0,0}(\phi, \theta)$ must form a partition of unity over the unit sphere

$$\sum_{n=1}^N U_n^{0,0}(\phi, \theta) \equiv 1 \quad (\text{B.3.6})$$

Assume that basis functions can be found satisfying conditions (B.3.3), (B.3.4) and (B.3.6). Then for functions $f(\phi, \theta)$ for which the above convergence condition is satisfied, (B.3.1) is a valid representation. This new representation uses the functional data at N sample points.

It is therefore a natural representation for use as an interpolation formula. We will call the truncated MT series

$$A_{\{M_1, \dots, M_N\}}[f](\phi, \theta) = \sum_{n=1}^N \sum_{0=k+l}^{M_n} f_n^{k,l} U_n^{k,l}(\phi, \theta) \quad (\text{B.3.7})$$

a MT interpolant of order $\{M_1, \dots, M_N\}$ for the function $f(\phi, \theta)$. It is readily clear that (B.3.7) is another and more natural solution to the Hermite-Birkhoff interpolation problem.

The main task is to construct basis function with the required properties. It turns out that such basis functions exist and even are not unique. This is in sharp contrast with the ordinary Taylor series representation where the basis functions are the unique familiar monomials. The non-uniqueness is no problem in the full MT series, since the exact form of the basis functions drop out if the summation is continued over all derivatives up to infinity. This is due to the partition of unity condition (B.3.6). But for the interpolant it is an additional freedom, which the algorithm designer can use to his advantage.

For the spherical basis functions we use the following radial construction. The choice (B.3.3) makes that we only have to consider zero order basis functions. The zero order basis function $U_n^{0,0}(\phi, \theta)$ associated to sample point $x_n = (\phi_n, \theta_n)$, will be constructed to have as finite support the Voronoi polygon V_n . To satisfy the partition of unity condition (B.3.6) we write

$$U_n^{0,0}(\phi, \theta) = \frac{\Phi_n(\phi, \theta)}{\sum_{i=1}^N \Phi_i(\phi, \theta)} \quad (\text{B.3.8})$$

For the fundamental functions $\Phi_n(x)$, $x = (\phi, \theta)$ we choose the form

$$\Phi_n(x) = \begin{cases} \Psi(w_n(x)), & x \in V_n \\ 0, & x \notin V_n \end{cases} \quad (\text{B.3.9})$$

This assures us of the local support of the basis functions $U_n^{0,0}(x)$ and hence $U_n^{k,l}(x)$. The real function $\Psi: \mathbb{R} \rightarrow \mathbb{R}$, such that $w \mapsto \Psi(w)$, should satisfy

$$\begin{aligned} \Psi(-\infty) &= 0 \\ \Psi(+\infty) &= 1 \\ \left. \frac{d^k \Psi(w)}{dw^k} \right|_{w \rightarrow \pm\infty} &= 0, \quad k > 0 \end{aligned} \quad (\text{B.3.10})$$

In addition, the real functions $w_n: \mathbb{S}^2 \rightarrow \mathbb{R}$, such that $x \mapsto w_n(x)$, should satisfy

$$w_n(x_p) = \begin{cases} +\infty, & p = n \\ -\infty, & p \neq n \end{cases} \quad (\text{B.3.11})$$

Conditions (B.3.10) and (B.3.11) will ultimately make that the $U_n^{k,l}(x)$ will satisfy the conditions (B.3.2). The detailed proof is beyond the scope of this report and will be published elsewhere. The general function $\Psi(w)$, called from its form a slope function, is independent of the data and can be chosen freely, as long as conditions (B.3.10) are satisfied. A possible choice is

$$\Psi(w) = \frac{1}{2} (1 + \exp(-\exp(-ew)) - \exp(-\exp(ew))), \quad e = \exp(1) \quad (\text{B.3.12})$$

The numerical evaluation of this function does not pose any problems if one takes some minor precautions to prevent overflow and underflow. Since it is independent of the data it could be tabulated to speed up the computations.

The functions $w_n(x)$ are obtained by the following radial construction. Let $x_n = (\phi_n, \theta_n)$ be the considered sample point and $x = (\phi, \theta) \neq x_n$ a point inside the spherical triangle $T_{a,n} \subset V_n$.

Consider the geodesic arc L , through x and x_n , and which intersects the convex Voronoi polygon boundary ∂V_n , at the points $\{x_a, x_b\} = L \cap \partial V_n$, such that $x_a \in T_a$ and $x_b \in T_b$. Map the arc segment $[x_a, x_b] \subset \mathbb{S}^2$ to the real line segment $[-a, b] \subset \mathbb{R}$, $a, b > 0$, such that $x \mapsto \omega$, $x_n \mapsto 0$, $x_a \mapsto -a$ and $x_b \mapsto b$. Define a real function $\hat{w}: \mathbb{R} \rightarrow \mathbb{R}$, such that $\omega \mapsto \hat{w}(\omega)$, by

$$\hat{w}(\omega) = \frac{7}{10} \frac{(a/2 + \omega)(b/2 - \omega)}{a+b} \left(\frac{c^-}{\omega+a} + \frac{1}{|\omega|} + \frac{c^+}{\omega-b} \right) \quad (\text{B.3.13})$$

The functions $w_n(x)$ are then defined as $w_n(x) = \hat{w}(\omega)$. The constants c^\pm in (B.3.13) are obtained from the conditions that

$$\left. \frac{d^2 \hat{w}(\omega)}{d\omega^2} \right|_{\omega=-a/2} = \left. \frac{d^2 \hat{w}(\omega)}{d\omega^2} \right|_{\omega=b/2} = 0 \quad (\text{B.3.14})$$

and are simple rational functions of a and b . The normalisation constant $7/10$ in (B.3.13) is taken over from the one-dimensional equidistant case [Franssens, 1998].

To summarise, the evaluation of a basis function proceeds as follows. Given an interpolation point x in the Voronoi polygon of a given sample point x_n , the geodesic passing through these two points determines intersection points x_a and x_b with the polygon boundary. The arc segment on this geodesic is mapped to the real line segment $-a \leq \omega \leq b$ and (B.3.13) is evaluated at ω , corresponding with x . This value is used as $w_n(x)$, which is then substituted in the slope function (B.3.12). Through (B.3.9) we get the corresponding value of the fundamental function $\Phi_n(x)$.

Notice that for any given interpolation point, only the three sample points that constitute the vertexes of the triangle containing the interpolation point, contribute. To evaluate a zero order basis function $U_n^{0,0}(x)$, it is therefore sufficient to evaluate only three fundamental functions in the denominator of (B.3.8). Values of higher order basis functions are then finally obtained from (B.3.3). This radial construction makes that a zero order basis function becomes zero on the Voronoi boundary and has all its radial derivatives zero there. It equals one at the interior sample point and has all its derivatives zero there as well.

A final remark about the basis functions and the analytic character of the resulting interpolant (B.3.7). The conditions (B.3.4) for the zero order basis functions show that the sample points are isolated essential singularities for these functions. This implies that the resulting interpolant has an isolated essential singularity at each sample point. However, as seen from conditions (B.3.4), the zero order basis functions are there well defined (being unity) and infinitely differentiable (having all derivatives zero). This means that the resulting interpolant is a $C^{\infty,\infty}$ function everywhere. However, it can not be expanded in (ordinary) Taylor series around any sample point (but could be given a Laurent series expansion there).

B.4 Estimation of spatial derivatives

The interpolant of (B.3.7) can take spatial derivative values into account in addition to the functional value. Since in practise derivatives are seldom available, we now describe a technique to estimate derivatives from local functional values. There are various ways to (numerically) estimate derivatives. The most obvious one is using finite difference formulas. However in a noise environment (i.e. functional values corrupted by measurement errors), this is a very error prone method. It could be used if the functional data are accurate and consistent with a smooth varying analytic function. Since this is not the case for aerosol data, with sometimes large error bars, we need a more robust technique. This has been found in the use of a local thin plate spline fit. The idea is to approximate the unknown function locally with a thin plate, bent in such a way that it passes to the sample values. By using the physical model of an elastic thin plate and determining its shape from the minimisation of its potential energy, one is assured that no wild oscillations or overshoots will be present in the resulting interpolant.

More precisely, we search around each sample point an optimal shape function $s_n : \mathbf{S}^2 \rightarrow \mathbf{R}$, such that $x \mapsto s_n(x)$, which minimises the following functional

$$\xi_n(s) = \delta_n(s) + \lambda \eta_n(s) \quad (\text{B.4.1})$$

wherein:

$$\delta_n(s) = \sum_{i=1}^{I_n} w_i (f_i - s_n(x_i))^2 \quad (\text{B.4.2})$$

is a weighted sample data matching cost function. The summation runs over the considered sample point x_n and its N_n Voronoi neighbours, $I_n = N_n + 1$. The numbers w_i are weights, which can take into account the error on the functional values f_i . The second term in (B.4.1) is a measure of the total curvature of the plate shape $s_n(x)$

$$\eta_n(s) = \int_{\mathbf{S}^2} \left[\left(\frac{\partial^2 s_n(\phi, \theta)}{\partial \theta^2} \right) + 2 \left(\frac{\partial^2 s_n(\phi, \theta)}{\partial \phi \partial \theta} \right) + \left(\frac{\partial^2 s_n(\phi, \theta)}{\partial \phi^2} \right) \right] d\Omega \quad (\text{B.4.3})$$

and λ can be called a fitting parameter. If $\lambda = 0$, minimisation of (B.4.1) is just (least square) interpolation. For $\lambda > 0$ the process is a thin plate fit.

The minimisation of (B.4.1) on the unit sphere is no trivial task. We will here simplify the problem by introducing a quasi-planar approximation. This means that we replace (B.4.3) by the planar functional

$$\eta_n(s) = \int_{\mathbf{R}^2} \left[\left(\frac{\partial^2 s_n(\phi, \theta)}{\partial \theta^2} \right) + 2 \left(\frac{\partial^2 s_n(\phi, \theta)}{\partial \phi \partial \theta} \right) + \left(\frac{\partial^2 s_n(\phi, \theta)}{\partial \phi^2} \right) \right] d\phi d\theta \quad (\text{B.4.4})$$

This approximation is justified since we are only interested in the plate shape in a small neighbourhood of the sample point (the Voronoi polygon). We thus replace the spherical metric with the local planar metric. This planar version of the variational problem (B.4.1) has the advantage that it has an analytic solution. For equal weights ($w = 1/I_n$) it has the form [Dierckx, 1995]:

$$s_n(\phi, \theta) = \sum_{i=1}^{I_n+3} a_i K_i(\phi, \theta) \quad (\text{B.4.5})$$

wherein

$$\begin{aligned} K_i(\phi, \theta) &= \frac{1}{16\pi} r_i^2(\phi, \theta) \log(r_i^2(\phi, \theta)), \quad 1 \leq i \leq I_n \\ K_{I_n+1}(\phi, \theta) &= 1 \\ K_{I_n+2}(\phi, \theta) &= \phi \\ K_{I_n+3}(\phi, \theta) &= \theta \\ r_i^2(\phi, \theta) &= (\phi - \phi_i)^2 + (\theta - \theta_i)^2 \end{aligned} \quad (\text{B.4.6})$$

The coefficients in (B.4.5) are determined from the linear system

$$\mathbf{Ma} = \mathbf{f} \quad (\text{B.4.7})$$

with

$$\mathbf{M} = \begin{bmatrix} \mathbf{K}(\lambda) & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} a_1 \\ \dots \\ a_{I_n} \\ a_{I_n+1} \\ a_{I_n+2} \\ a_{I_n+3} \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} f_1 \\ \dots \\ f_{I_n} \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{B.4.8})$$

$$\mathbf{K}(\lambda) = \begin{bmatrix} K_1(\phi_1, \theta_1) + I_n \lambda & \dots & K_{I_n}(\phi_1, \theta_1) \\ \dots & \dots & \dots \\ K_1(\phi_{I_n}, \theta_{I_n}) & \dots & K_{I_n}(\phi_{I_n}, \theta_{I_n}) + I_n \lambda \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 1 & \phi_1 & \theta_1 \\ \dots & \dots & \dots \\ 1 & \phi_{I_n} & \theta_{I_n} \end{bmatrix} \quad (\text{B.4.9})$$

In our application, the dimension of the linear system is typical about 10 by 10. Numerical instabilities are reported in the literature for these kind of systems, but only for larger dimensions (above 100 x 100). For the small dimensions we use, numerical problems were never encountered.

Using the derivatives of the thin plate shape in (B.3.7) is equivalent to using as interpolant

$$\begin{aligned} A[f](\phi, \theta) &= \sum_{n=1}^N \sum_{0=k+l}^{+\infty} s_n^{k,l} U_n^{k,l}(\phi, \theta) \\ &= \sum_{n=1}^N \left(\sum_{0=k+l}^{+\infty} s_n^{k,l} \frac{(\phi - \phi_n)^k}{k!} \frac{(\theta - \theta_n)^l}{l!} \right) U_n^{0,0}(\phi, \theta) \\ &= \sum_{n=1}^N s_n(\phi, \theta) U_n^{0,0}(\phi, \theta) \end{aligned} \quad (\text{B.4.10})$$

Since the support of the basis functions is limited to its Voronoi polygon, only a local portion of the thin plate shape is used. Moreover, any interpolated value is a linear combination of the three thin plate shapes associated to the three vertexes of the triangle in which the interpolation point is located. So for any point $x = (\phi, \theta)$ the sum in (B.4.10) is reduced to three terms. These three shapes are blended together by the zero order MT basis functions. Again the interpolant (B.4.10) has the same analyticity properties as mentioned higher. This is the formula implemented in the final algorithm.

B.5 Final spatial filtering

After the interpolation, as computed by (B.4.10), the user has the possibility to do an additional smoothing operation. Smoothing can be applied over longitudes or over latitudes. Its main purpose is to remove unphysical variations in the map, having a higher spatial frequency than the Nyquist sampling frequency, determined by the satellite orbit and mode of operation characteristics.

Smoothing is done by applying a low pass, spatial frequency filter, of Gaussian shape. Filtering is done by taking the FFT of the unfiltered map, then multiplying it by the filter

$$g(f_\phi, f_\theta) = \exp\left(-\left(f_\phi / f_a\right)^2\right) \exp\left(-\left(f_\theta / f_b\right)^2\right) \quad (\text{B.5.1})$$

and inverse FFT the product. Herein is f_a the longitudinal spatial frequency and f_b a filter characteristic which determines the spectral width of the filter and is controlled by the user.

B.6 File format definitions

B.6.1 Format 'PRF'

The file format 'prf' is used to hold 3D data arranged in vertical profiles located at sampling points randomly scattered over the globe. In this format it is assumed that the altitude sampling is the same at each surface sampling point. For data sources for which this is not the case altitude interpolation towards common altitudes is used.

1. Files

The format 'prf' data file set consists of the following files:

name.prf.nsp:	number of sampling points
name.prf.lon:	longitudes of the sampling points
name.prf.lat:	latitudes of the sampling points
name.prf.alt:	altitudes where the profiles are sampled
name.prf.tpp:	tropopause altitudes at the sampling points
name.prf.mva:	minimum valid altitudes at the sampling points
name.prf.val:	the 3D set of extinction (beta) values (in 1/km)
name.prf.err:	the 3D set of absolute errors on beta values (in 1/km)
name.prf.prs:	the 3D set of pressure values (in mbar)
name.prf.tmp:	the 3D set of temperature values (in degree K)

The 'name' is the user's choice, the extensions are obligatory.

2. Formats

Data is written to these files in formatted (ASCII) form to ensure portability across platforms. The record structure for each file is as follows:

name.prf.nsp

Structure: number_of_profiles <TAB> number_of_altitudes <NL>

Type: long number_of_profiles, number_of_altitudes;
Written as: “%d”

name.prf.lon

Structure: longitude_1 <TAB> longitude_2 <TAB> ... longitude_number_of_profiles <NL>

Type: float longitude_n;
Written as: “%e”

name.prf.lat

Structure: latitude_1<TAB> latitude_2<TAB>... latitude_number_of_profiles <NL>

Type: float latitude_n;
Written as: “%e”

name.prf.alt

Structure: altitude_1<TAB> altitude_2<TAB>... altitude_number_of_altitudes<NL>

Type: float altitude_n;
Written as: “%e”

name.prf.mva

Structure: altitude_1<TAB> altitude_2<TAB>... altitude_number_of_profiles<NL>

Type: float altitude_n;
Written as: “%e”

name.prf.tpp

Structure: altitude_1<TAB> altitude_2<TAB>... altitude_number_of_profiles<NL>

Type: float altitude_n;
Written as: “%e”

name.prf.val

Structure: profile_1< NL > profile_2< NL >... profile_number_of_profiles<NL>
wherein profile_n = value_1<TAB> value_2<TAB>... value_number_of_altitudes

Type: float value_n;
Written as: “%e”

name.prf.err

Structure: err_prf_1< NL > err_prf_2< NL >... err_prf_number_of_profiles<NL>
wherein err_prf_n = error_1<TAB> error_2<TAB>... error_number_of_altitudes

Type: float error_n;
Written as: “%e”

name.prf.prs

Structure: profile_1< NL > profile_2< NL >... profile_number_of_profiles<NL>
wherein profile_n = value_1<TAB> value_2<TAB>... value_number_of_altitudes

Type: float value_n;
Written as: “%e”

name.prf.tmp

Structure: profile_1<NL> profile_2<NL>... profile_number_of_profiles<NL>
wherein profile_n = value_1<TAB> value_2<TAB>... value_number_of_altitudes

Type: float value_n;
Written as: "%e"

Herein are <TAB> = "\t" and <NL> = "\n".

B.6.2 Format 'ACT'

The file format 'act' is used to hold 2D data produced by intermediate processors (see Chapter 3) acting on the 'prf' data set.

Examples of this kind of data are an altitude section, total column extinction, isobar surface, isentropic surface, etc. The 'act' string reflects the process used to produce the data, e.g. 'act' = 'tce' for total column extinction data.

1. Files

The format 'act' data file set consists of the following files:

name.act.nsp:	number of sampling points
name.act.lon:	longitudes of the sampling points
name.act.lat:	latitudes of the sampling points
name.act.val:	the 2D set of sampling values
name.act.err:	the 2D set of absolute errors on the sampling values

2. Formats

The file formats are the same as for the 'sam' set described below.

B.6.3 FORMATS 'SAM' AND 'INT'

The file formats 'sam' and 'int' are used to hold 2D input and output data respectively for the interpolation algorithm.

1. Files

The input format 'sam' data file set consists of the following files:

name.sam.nsp:	number of sampling points
name.sam.lon:	longitudes of the sampling points
name.sam.lat:	latitudes of the sampling points
name.sam.val:	the 2D set of sampling values
name.sam.err:	the 2D set of absolute errors on the sampling values

The output format 'int' data file set consists of the following files:

name.int.nsp:	number of longitude and latitude grid points
name.int.lon:	longitudes of the interpolation grid
name.int.lat:	latitudes of the interpolation grid
name.int.val:	the 2D set of interpolated values

name.int.err: the 2D absolute error estimate on the interpolated values

The 'name' is the user's choice, the extensions are obligatory.

Two additional files are optionally made:

name.sam.gif: a gif type picture file showing the sample points

name.int.gif: a gif type picture file showing the interpolated map

The 'name' may be of a composed nature, e.g. 'JAN_92.tce', standing for the total column extinction data for the January 1992 period.

2. Formats

Data is written to these files in formatted (ASCII) form to ensure portability across platforms. The record structure for each file is as follows:

a) Input ('sam')

name.sam.nsp

Structure: number_of_samples<NL>

Type: long number_of_samples;

Written as: "%d"

name.sam.lon

Structure: longitude_1<TAB> longitude_2<TAB>... longitude_number_of_samples<NL>

Type: float longitude_n;

Written as: "%e"

name.sam.lat

Structure: latitude_1<TAB> latitude_2<TAB>... latitude_number_of_samples<NL>

Type: float latitude_n;

Written as: "%e"

name.sam.val

Structure: value_1<TAB> value_2<TAB>... value_number_of_samples<NL>

Type: float value_n;

Written as: "%e"

name.sam.err

Structure: error_1<NL> error_2<NL>... error_number_of_samples<NL>

Type: float error_n;

Written as: “%e”

b) Output (‘int’)

name.int.nsp

Structure: number_of_longitudes<TAB>number_of_latitudes<NL>

Type: long number_of_samples, number_of_latitudes;

Written as: “%d”

name.int.lon

Structure: longitude_1<TAB> longitude_2<TAB>... longitude_number_of_longitudes<NL>

Type: float longitude_n;

Written as: “%e”

name.int.lat

Structure: latitude_1<TAB> latitude_2<TAB>... latitude_number_of_latitudes<NL>

Type: float latitude_n;

Written as: “%e”

name.int.val

Structure: zone_1<NL> zone_2<NL>... zone_number_of_latitudes<NL>

wherein zone_n = value_1<TAB> value_2<TAB>... value_number_of_longitudes

Type: float value_n;

Written as: “%e”

name.int.err

Structure: zone_1<NL> zone_2<NL>... zone_number_of_latitudes<NL>

wherein zone_n = error_1<TAB> error_2<TAB>... error_number_of_longitudes

Type: float error_n;

Written as: “%e”

Herein are <TAB> = “\t” and <NL> = “\n”.