

Qpack, a general tool for instrument simulation and retrieval work

Patrick Eriksson^{a,*}, Carlos Jiménez^a, Stefan A. Buehler^b

^a*Department of Radio and Space Science, Chalmers University of Technology, SE-412 96 Göteborg, Sweden*

^b*Institute of Environmental Physics, University of Bremen, P.O. Box 330440, D-28334 Bremen, Germany*

Received 30 July 2003; accepted 21 May 2004

Abstract

Remote sensing requires a complete observation system, consisting of the instrument, a forward model and a retrieval environment. This paper presents a software tool to complement atmospheric sensors, with focus on passive instruments operating in the mm and sub-mm wavelength regions. The tool is of general character and offers a complete, flexible and fast calculation environment, demonstrated in both preparatory instrument studies and operational inversions. Its features include a rapid approach for modelling of sensor characteristics, several types of data reduction, simple definition of covariance matrices, a large number of retrieval and error quantities, inversion characterisation and random realisation of measurements. The software is freely available for scientific use.

© 2004 Elsevier Ltd. All rights reserved.

PACS: 07.05.Kf; 07.07.Df; 93.85.+q; 92.70.Cp

Keywords: Remote sensing; Atmosphere; Radiative transfer; Sensor modelling; Inversion methodology; Software

1. Introduction

Remote sensing is an important technique for measuring various atmospheric quantities. For example, the use of satellite based sensors is steadily increasing, both for basic research and

*Corresponding author. Fax: +46-31-7721884.

E-mail address: patrick.eriksson@rss.chalmers.se (P. Eriksson).

weather forecasting. The indirect nature of the observations requires that, beside the actual instrument, a forward model and a retrieval environment are at hand. These three components can together be treated as the indispensable parts of the observation system [1]. The forward model is the computerised tool used to simulate atmospheric radiative transfer and sensor characteristics. The process of extracting geophysical parameters from observed spectra is denoted as the retrieval, or the inversion.

Buehler et al. (this issue) deals with ARTS, an extensive and modular software, written in C++, for simulating atmospheric radiative transfer. This paper describes an accompanying Matlab package, Qpack, that together with ARTS constitutes a complete and general environment for forward modelling and retrieval work. The package contains functions for sensor modelling, data reduction, inversion of observed spectra, error characterisation, optimisation of calculation grids and random realisation of measurements. Qpack was developed to be as general as possible and the applications span from ground-based mm-wave radiometry [2] to limb radio occultation [3]. In addition, ARTS and Qpack are used in connection with the sub-mm limb sounding measurements by the Odin satellite [4], both for the operational retrievals and a secondary inversion chain based on neural nets [5].

It is not possible to give here a complete description of Qpack. The aim is instead to give an overview of Qpack. In addition, some of the solutions developed to make Qpack such a flexible tool are presented, as these features are probably unique for Qpack. However, a comparison to other similar inversion codes is difficult as they are not described in the open literature, one exception is [6]. Qpack is outstanding also in this respect, the tool is not only described here, it is also available freely for scientific use (Section 3.2).

2. Theory

2.1. A theoretical formalism

The forward model, F , is here defined as [7,1]

$$\mathbf{y} = F(\mathbf{x}, \mathbf{b}) + \varepsilon, \quad (1)$$

where \mathbf{y} is the vector of measurement data, the vector \mathbf{x} holds the variables to be retrieved, \mathbf{b} is the vector of remaining forward model parameters and ε is the measurement noise. The forward model has two main sections; a first part describing atmospheric radiative transfer,

$$\mathbf{i} = F_r(\mathbf{x}_r, \mathbf{b}_r), \quad (2)$$

and a second part where sensor characteristics are treated,

$$\mathbf{y} = F_s(\mathbf{i}, \mathbf{x}_s, \mathbf{b}_s) + \varepsilon, \quad (3)$$

where \mathbf{i} is a vector holding monochromatic pencil beam spectral values, and the vectors \mathbf{x} and \mathbf{b} are divided each into two parts, following the division of the forward model. The term sensor characteristics also covers here a possible data reduction, to transform \mathbf{y} into a space of smaller dimension.

In [8] it is pointed out that the sensor part of the forward model can be implemented by a multiplication between a response matrix, \mathbf{H} , and \mathbf{i} . That is,

$$\mathbf{y} = \mathbf{H}\mathbf{i} + \varepsilon, \quad (4)$$

on the condition that all sensor responses and data reduction transformations are linear operations. See further Section 4.

The total retrieval error, δ , for inversion cases with a not too strong non-linearity, can be calculated as

$$\delta = \hat{\mathbf{x}} - \mathbf{x} = (\mathbf{A} - \mathbf{I})(\mathbf{x} - \mathbf{x}_a) + \mathbf{D}_y[\mathbf{K}_b(\mathbf{b} - \mathbf{b}_a) + \varepsilon], \quad (5)$$

where $\hat{\mathbf{x}}$ is the retrieved state, $\mathbf{A} = \partial\hat{\mathbf{x}}/\partial\mathbf{x}$, \mathbf{I} the identity matrix, \mathbf{x}_a and \mathbf{b}_a are the a priori estimate of \mathbf{x} and \mathbf{b} , respectively, $\mathbf{D}_y = \partial\hat{\mathbf{x}}/\partial\mathbf{y}$, and $\mathbf{K}_b = \partial F/\partial\mathbf{b}$. See further [7,1]. As argued in [9], the distinction between forward model uncertainties ($\mathbf{b} - \mathbf{b}_a$) and measurement errors (ε) is vague, and the two terms are here denoted together as the observation error, \mathbf{e} (following the idea of an observation system):

$$\mathbf{e} = \mathbf{K}_b(\mathbf{b} - \mathbf{b}_a) + \varepsilon. \quad (6)$$

As the exact values of \mathbf{x} , \mathbf{b} and ε are not known, the retrieval error can only be estimated on statistical basis, using covariance matrices (\mathbf{S}):

$$\mathbf{S}_\delta = (\mathbf{A} - \mathbf{I})\mathbf{S}_x(\mathbf{A} - \mathbf{I})^T + \mathbf{D}_y\mathbf{S}_e\mathbf{D}_y^T, \quad (7)$$

where $\mathbf{S}_e = \mathbf{K}_b\mathbf{S}_b\mathbf{K}_b^T + \mathbf{S}_\varepsilon$, \mathbf{S}_x describes statistically $(\mathbf{x} - \mathbf{x}_a)$, and \mathbf{S}_δ , \mathbf{S}_b and \mathbf{S}_ε are defined likewise. See [9] for a more detailed discussion on how to interpret the covariance matrices in this context.

2.2. OEM, the optimal estimation method

The formalism presented in Section 2.1 is of general character and is independent of the retrieval method used, but limited to linear and moderately nonlinear inversion cases. The inversion method is characterised by the matrix \mathbf{D}_y . The most common retrieval approach for atmospheric soundings is statistical regularisation, normally denoted as OEM [10]. For linear inversions, OEM can be expressed as

$$\mathbf{D}_y = (\mathbf{K}_x^T\mathbf{S}_e^{-1}\mathbf{K}_x + \mathbf{S}_x^{-1})^{-1}\mathbf{K}_x^T\mathbf{S}_e^{-1}, \quad (8)$$

where $\mathbf{K}_x = \partial F/\partial\mathbf{x}$. See [1] for an extensive discussion around OEM.

3. Qpack

3.1. Scope

The basic objective of developing Qpack was to create a shell around the ARTS program (Buehler et al., this issue) to obtain a complete tool for forward modelling and retrieval work. The stable branch of ARTS (1-0- x) is limited to atmospheric radiative transfer, and functionality for sensor modelling and inversions had to be added. The development of Qpack had no specific

instrument in mind, on the contrary, the aim was to obtain a high order of generality. As in ARTS, any observation geometry is allowed. However, as no features regarding atmospheric radiative transfer are added, Qpack is limited to cases that can be handled with ARTS (see Buehler et al., this issue). Focus is put on observations at mm and sub-mm wavelengths, as for ARTS, but the extension to other measurement techniques was a main consideration when creating Qpack. Anyhow, instrument designs do not fall into a single template and a total generality cannot be achieved. This means that the modelling of sensor characteristics often requires special solutions, and Qpack allows a simple solution to this problem (see Section 3.3). This can be compared to a retrieval environment where a very high degree of generality can be achieved. So far only inversions by OEM (Section 2.2) are supported.

Qpack can be used in various ways. For example, it can be used to just perform forward model calculations, this including generation of measurement data sets based on random inputs with prescribed statistics (Section 7). Furthermore, the retrieval environment offers a high flexibility and capability for both preparatory instrument studies and operational inversions. However, Qpack acts here normally only as the computational core. This is the case as the retrieval part of Qpack can only perform individual inversions. It was judged that an acceptable general solution of batch inversions would be vary hard to achieve. On the other hand, the well defined and flexible input of Qpack makes it easy to develop additional code that solves this problem for particular cases, a statement supported by the successful application of Qpack on operational inversions of sub-mm spectra measured by the Odin satellite.

3.2. Organisation

The Matlab environment around ARTS-1-0-*x* is divided into two parts, AMI and Qpack. AMI (ARTS Matlab interface) was created with the aim of making ARTS a complete forward model, and it thus contains the basic parts for sensor modelling. AMI further holds the functions for reading and writing of ARTS input and output files. All functions of general characters are also placed in AMI. Functions found inside Qpack are specific for the package, and can easily be recognised as their names all start with “qp” (for example, qp_H.m). However, no distinction will here be made between AMI and Qpack, and the two parts are together denoted as Qpack. AMI is distributed with ARTS. Qpack, like ARTS, is distributed following the GNU general public license, and can be downloaded from www.sat.uni-bremen.de/arts/. The main sources of documentation, beside this paper, are the README file coming with Qpack, the sample files found in the folder `samples`, and the on-line function descriptions (obtained by the Matlab `help` command).

3.3. Implementation strategy

The package is built up around a setting structure denoted as `Q`, which explains the name Qpack. The `Q` structure has a set of prescribed fields (listed and described in the README file), but additional fields can be added for particular purposes. The standard way to set `Q` is creating a function that has `Q` as the only input and output arguments (for example, `Q = sample_q(Q)`). To ensure that all prescribed fields are set, the `Q` definition function (QDF) shall be called through the function `qpack` (for example, `Q = qpack('sample_q')`). The function `qpack` starts by setting

default values where possible, followed by a call of the QDF. The output will be the combination of default settings and the choices in the QDF, where the settings in QDF have the highest priority.

All internal Matlab functionality can be used to set the Q structure. For example, the QDF can contain calculations and function calls to set the fields of Q. A very useful solution is to use a hierarchy of QDFs. This approach permits a set of simulation cases to be created without having to duplicate information, a field common for all cases is set in the lowermost QDF and so on. Another common solution for preparatory studies is to make functions that take a set Q as input and vary one or several fields in a systematic way, to investigate the inversion performance for a range of values. The calculations can in this way be repeated very simply if any of the other settings is changed.

The quantities \mathbf{H} , \mathbf{S}_e and \mathbf{S}_x (Section 2) are treated specially, for reasons of efficiency and flexibility. For simulation conditions fully covered by Qpack, those quantities are pre-calculated by the function calls `qp_H(Q)`, `qp_Se(Q)` and `qp_Sx(Q)`, respectively. If only forward model calculations are to be performed (no inversion), only \mathbf{H} has to be generated. The pre-calculation of these quantities adds to the flexibility of Qpack. If the simulations around a specific instrument are not completely handled by the standard functionality of Qpack, this can often be solved by replacing `qp_H(Q)` and/or `qp_Se(Q)` with instrument specific version of those functions. For example, the ground-based CO observations reported in [2] were obtained by frequency switching, a detection technique not covered by Qpack, and this was solved by solely making a special version of `qp_H(Q)`. Simulation of limb radio occultation measurements [3], an application not envisaged when designing Qpack, was managed by creating a dedicated version of `qp_Se(Q)`.

The measurement spectrum corresponding to a specified Q structure is obtained as $\mathbf{y} = \text{qp_y}(Q)$, on the condition that \mathbf{H} is pre-calculated. A simulated or observed measurement, \mathbf{y} , is inverted as `qp_cls(Q, y)`, on the condition that \mathbf{H} , \mathbf{S}_e and \mathbf{S}_x are pre-calculated. Type `help qp_cls` for a description of output from the function. A general error characterisation is obtained as `qp_cls_invhar(Q)`, while individual error sources are investigated by `qp_cls_errorplot(Q)` (see further Section 6.3). See Section 7 for random realisation of measurements.

4. Sensor modelling and data reduction

The rapid approach for inclusion of sensor characteristics and data reduction described by Eq. (4), was suggested by [8] but became first implemented in conjunction with ARTS-1-0-x. The idea is based on the fact that the response of the different sensor parts corresponds typically to some linear operation. The same is normally also true for data reduction techniques. This means that sensor responses, and data reduction, can be implemented by a single response matrix, \mathbf{H} . If several sensor parts must be considered, maybe together with a data reduction, the total response matrix is obtained as

$$\mathbf{H} = \mathbf{H}_n \dots \mathbf{H}_2 \mathbf{H}_1, \quad (9)$$

where \mathbf{H}_i is the matrix for each individual response considered. The derivation of \mathbf{H} for different sensor parts and type of data reduction is described in the ARTS User Guide (distributed with the ARTS source code), and will be the topic of a forthcoming paper.

Qpack covers the following sensor and data reduction options:

Antenna [ANTENNA_ON]. The angular response of the antenna, or the aperture, to weight pencil beam spectra from different directions. This response is described by giving the centre zenith angle for each spectrum (ANTENNA_ZA) and the antenna response around these directions (ANTENNA_FILE). The response is treated to be identical for all spectra. An antenna in constant motion can be considered (ANTENNA_MOVE).

Mixer and sideband [DSB_ON]. Mm and sub-mm receivers have normally sensitivity to two sidebands. The weighting between the bands (DSB_FILE) can be considered with this option. The local oscillator frequency (DSB_LO) and a frequency of the primary band must be specified (DSB_FPRIMARY).

Spectrometer [BACKEND_ON]. The response of the spectrometer, often denoted as the backend. The complete response is specified by giving the centre position of each backend channel (BACKEND_FREQS) and the response around these frequencies (BACKEND_FILE). All spectrometer channels are assumed to have the same response.

Spectral averages. Spectral values can be averaged to reduce the data size, often denoted as binning. Both subsequent (complete) spectra [BINVIEW_ON] and neighbouring spectrometer channels [BINNING_ON] can be binned.

Eigenvector reduction. Data reduction based on an eigenvector expansion of the measurement space, as described in [11]. The reduction can either be based on the complete spectral space [KRED_ON], or only on a part of a limb scanning sequence [LRED_ON].

The master Qpack field for each option is given inside square brackets. The option is activated by setting the master field to 1 (or any other non-zero value). Default is throughout 0. Connected sensor fields are given above (inside parentheses), while a complete list on both sensor and data reduction fields is found in the README file, where also more documentation is given. If the master field is set to 0, the connected fields can be left out from Q.

All the sensor responses are specified as files. The response files include a matrix of 2 columns, where column 1 is position and column 2 is the response at those positions. The response for intermediate positions is linearly interpolated. Antenna and backend responses are specified with respect to the centre positions given (that is, on a grid centred around zero), while the sideband filter responses are given for absolute frequencies. The responses are properly normalised inside Qpack, and the files can contain any common (linear) scaling of the responses.

5. Definition of covariance matrices

Both the retrieval characterisation (Eq. (7)) and the actual inversion (Eq. (8)) are built up around the use of covariance matrices. It is accordingly a requirement that covariance matrices can be specified, but there is often a lack of statistical information, at least for the correlation between different members of the state (\mathbf{x}) and forward model (\mathbf{b}) vectors. This lack of information results in that the covariance matrices are often set to have only diagonal elements, implying a zero correlation. However, the optimal situation is that the covariance matrices reflect the true situation, and it is more common to have considerable correlation rather than no correlation at all, at least between the values describing each retrieval quantity, such as the vertical

profile of a gas or temperature (see for example [12]). To neglect existing correlation can have significant negative impact on the inversion accuracy, as shown by [9].

Sufficient external data does not normally exist to calculate the covariance matrices numerically. There could exist partial data, but it is not straightforward to extend a numerical covariance matrix, as for theoretical reasons it should be positive definite. A part of this requirement is that it must be possible to invert a covariance matrix, as for OEM (Eq. (8)), which is not ensured by manually extending a matrix numerically derived. For example, sonde data can be used to derive complete ozone statistics up to about 30 km [12], but information for higher altitudes is also normally needed. The approach used for Qpack to solve this practical problem is to use a study such as [12] to estimate the standard deviation and a descriptive value for the correlation, and to use these values for a parameterised description of the covariance matrix. The correlation parameter is the length to the point where the correlation has declined to $\exp(-1)$, denoted as the correlation length (unit depending on quantity described). The standard deviation and the correlation length are then specified in the input file for a set of positions. When finally creating the covariance matrix, values for intermediate positions are linearly interpolated.

The correlation is further specified by selecting a functional type. The covariance matrix, \mathbf{S} , with a Gaussian correlation function is

$$\mathbf{S}(i,j) = \sigma(i)\sigma(j) \exp\left(-4\left(\frac{z(i) - z(j)}{l_c(i) + l_c(j)}\right)^2\right), \quad (10)$$

where i and j are position indexes, σ is the standard deviation, z is the position and l_c is the correlation length. It should be noted that the mean between the correlation lengths at the two positions must be used, as done in Eq. (10), to fulfil the symmetry demand on covariance matrices. The correlation function can also be exponential,

$$\mathbf{S}(i,j) = \sigma(i)\sigma(j) \exp\left(\frac{-2|z(i) - z(j)|}{l_c(i) + l_c(j)}\right), \quad (11)$$

where $|\cdot|$ signifies the absolute value, or be linearly decreasing (a tent function)

$$\mathbf{S}(i,j) = \max\left\{0, \sigma(i)\sigma(j) \left[1 - (1 - e^{-1}) \frac{2|z(i) - z(j)|}{l_c(i) + l_c(j)}\right]\right\}. \quad (12)$$

In addition, the covariance matrix can be set to be strictly diagonal. The correlation length values are then ignored.

The Gaussian and exponential options fill a correlation matrix with non-zero values, which is disadvantageous both with respect to memory usage (the covariance matrices are stored as sparse matrices) and numerical problems when inverting the matrix. This problem is solved by introducing a correlation cut-off, all correlations (that is, Eqs. (10)–(12) with the term $\sigma(i)\sigma(j)$ removed) below the given threshold are set to 0.

The covariance matrix for each retrieval and error quantity (see Section 6) is specified separately. All correlations between different quantities are set to zero when creating \mathbf{S}_x and \mathbf{S}_e . The matrices are defined by a file format. A description is obtained on-line by executing `help sFromFile`. The file format allows that the covariance matrix can be described as a sum of matrices ($\mathbf{S} = \mathbf{S}_1 + \mathbf{S}_2 + \dots$), where all parameters described above can be specified individually

for each matrix. This option is of interest when the variability (or uncertainty) of a variable is governed by several uncorrelated processes. For example, there are at least two terms to consider when describing the uncertainty of the distribution of a gas or temperature: the uncertainty caused by the dynamic and chemistry of the atmosphere, and the measurement and/or model errors giving an error of assumed mean state [9]. If the two terms are estimated using different data sources, it should be reasonable that the corresponding uncertainties are uncorrelated.

6. Retrievals and error characterisation

6.1. Retrieval and error quantities

A large number of variables affect a remote sensing measurement. It is not common to simultaneously retrieve all of those variables. For example, systematic forward model uncertainties are normally not considered during regular inversions. Attempts to fit systematic parameters are then done off-line, by dedicated inversion efforts. Another option is to treat a quantity as an observation uncertainty (see further Section 8). Accordingly, in a good retrieval environment it must be possible to treat the variables in different ways depending on the conditions and purpose of the inversions. In Qpack this is achieved by setting the “do-level”. The levels have a number code:

Level 3. The quantity is retrieved and is then throughout part of the state vector \mathbf{x} .

Level 2. The quantity is treated as an observation uncertainty. That is, it is throughout part of either \mathbf{b} or ε .

Level 1. Consider the quantity only for dedicated error characterisations (by `qpcls_error_plot`). Hence, the quantity is part of either \mathbf{b} or ε for Eq. (7) but is ignored during the actual inversion by Eq. (8).

Level 0. Ignore, or treat the quantity as perfectly known.

For example, `Q.POINTING_DO = 3` means that a pointing off-set (see below) shall be retrieved. The default level is throughout 0.

The following retrieval and error quantities are handled by Qpack (where the master field is given inside square brackets):

Measurement thermal noise [`MEASNOISE_DO`]. Thermal noise uncorrelated between spectra. Do-level 3 is not specified for thermal noise. Unit is K for cases with emission and optical thickness for pure transmission cases.

Calibration thermal noise [`CALINOISE_DO`]. Thermal noise totally correlated between spectra. Do-levels and unit as above.

Species Atmospheric species. An arbitrary number (including zero) of species can be retrieved. Species listed in the field `RETRIEVAL_TAGS` are treated as retrieval quantities (level 3), while species found in the field `OTHER_TAGS` are included in the calculations assuming a perfect knowledge on the abundance (level 0). Species are retrieved in fractions of the a priori profile. That is, 1.1 means that the abundance is 10% above the a priori value.

Temperature [`TEMPERATURE_DO`]. The vertical profile of atmospheric temperatures. Unit is K.

- Pointing off-set** [POINTING_DO]. A zenith angle correction term that is common for all spectra. Unit is degree.
- Frequency off-set** [FREQUENCY_DO]. A frequency correction term that is common for all spectrometer channels. Unit is Hz.
- Continuum absorption** [CONTABS_DO]. Absorption that varies smoothly with frequency. The polynomial representation used is described in the ARTS User Guide. The continuum absorption can be limited to a specified frequency range, such as the primary observation band. The standard deviation is specified by selecting a set of species and a relative variability. The final standard deviation is then obtained by calculating the summed absorption of the species and multiplying the result with the relative variability.
- Ground emission** [EGROUND_DO]. The emission factor of the ground. The emission factor can either be defined to be common for all frequencies, or to have a frequency variation. Retrieved values can be bound to any range, where [0, 1] is the natural choice. Non-dimensionless unit.
- Polynomial additive term** [POLYFIT_DO]. Contribution to the spectra that can be modelled by adding a polynomial term to each spectrum (a polynomial baseline shift). The polynomial order is arbitrary. There is a polynomial connected to each spectrum. The covariance matrices describe the statistics for each polynomial coefficient. Unit as for thermal noise.
- Piecewise polynomial additive term** [PPOLYFIT_DO]. As above, but with a baseline shift modelled as piecewise polynomials. The two types of polynomial fits can be combined.
- Calibration load temperatures** [TB_REFLOADS_DO]. The effective brightness temperature of calibration loads. A load switching procedure is assumed, involving two loads at different temperatures. Unit is K.
- Proportional calibration error** [PROPCAL_DO]. A scaling factor that is common for all spectra. For example, an incorrect correction of the tropospheric attenuation falls into this category for ground-based observations of the middle atmosphere. Non-dimensionless unit.

For all quantities, see further the README file distributed with Qpack. For scalar quantities (pointing/frequency off-set and calibration variables) the variability/uncertainty is specified by giving a standard deviation. For all remaining variables it is possible to define covariance matrices following the scheme described in Section 5. Standard deviations are given in the units given above. The correlation length unit is frequency for thermal noise and the ground emission factor, and zenith angle for polynomial fit variables. For species, temperature and continuum absorption the base 10 logarithm of the pressure is used as altitude coordinate as the correlation length should be much more constant in geometrical altitude than in pressure. One unit step in log₁₀ pressure corresponds to roughly 16 km throughout the atmosphere. The scheme used allows, for example, describing the correlation of thermal noise between neighbouring spectrometer channels, and the correlation of baseline fit parameters between subsequent spectra.

6.2. Retrievals

As mentioned, OEM is the only inversion method yet covered. An extension to also handle Tikhonov regularisation (TR) has been considered, but is not implemented. Such an extension would be straightforward due to the similarities between OEM and TR [9]. This considered

extension is reflected in Qpack by the fact that OEM must be explicitly selected (RETRIEVAL_METHOD) and OEM is presented as a constrained least squares (CLS) method, a term also covering TR. The function for performing inversions is accordingly called `qpcls`.

Linear or nonlinear retrievals are selected by the field `CLS_NONLIN_ON`. Linear retrievals follow Eq. (8). Nonlinear cases require an iterative search for the solution where the Marquardt–Levenberg approach is used. The used iteration equation is [1]

$$\mathbf{x}_{i+1} = \mathbf{x}_i + (\mathbf{K}_i^T \mathbf{S}_e^{-1} \mathbf{K}_i + \mathbf{S}_x^{-1} + \gamma \mathbf{D}_x^{-1})^{-1} [\mathbf{K}_i^T \mathbf{S}_e^{-1} (\mathbf{y} - F(\mathbf{x}_i, \mathbf{b})) - \mathbf{S}_x^{-1} (\mathbf{x}_i - \mathbf{x}_a)], \quad (13)$$

where \mathbf{x}_i is the state vector at iteration i , \mathbf{K}_i is \mathbf{K}_x evaluated at \mathbf{x}_i , γ is a trade-off parameter, and \mathbf{D}_x equals \mathbf{S}_x but with off-diagonal elements set to zero. The cost function to minimise by Eq. (13) is

$$[\mathbf{y} - F(\mathbf{x}_i, \mathbf{b})]^T \mathbf{S}_e^{-1} [\mathbf{y} - F(\mathbf{x}_i, \mathbf{b})] + [\mathbf{x}_i - \mathbf{x}_a]^T \mathbf{S}_x^{-1} [\mathbf{x}_i - \mathbf{x}_a]. \quad (14)$$

The iteration is initiated by setting \mathbf{x} to \mathbf{x}_a . The start value of γ is given by the field `CLS_GA_START_VALUE`. If the cost function for \mathbf{x}_{i+1} is smaller than for \mathbf{x}_i , the new state is accepted and γ is decreased with the factor in field `CLS_GA_FAC_WHEN_OK`. Otherwise, γ is increased with the factor in field `CLS_GA_FAC_WHEN_NOT_OK` and the iteration is retried. The iteration is stopped when a successful iteration gives

$$\frac{(\mathbf{x}_{i+1} - \mathbf{x}_i)^T \mathbf{S}_\delta^{-1} (\mathbf{x}_{i+1} - \mathbf{x}_i)}{n} < \Delta x, \quad (15)$$

where n is the length of \mathbf{x} and Δx is specified by the field `CLS_STOP`. For OEM \mathbf{S}_δ^{-1} can be approximated as

$$\mathbf{S}_\delta^{-1} = \mathbf{K}_i^T \mathbf{S}_e^{-1} \mathbf{K}_i + \mathbf{S}_x^{-1}. \quad (16)$$

Computational time is saved by noting that the right hand side of Eq. (16) is also part of Eq. (13). The scaling by n in Eq. (15) has the advantage that Δx can be kept constant independent of the quantities and grids selected for \mathbf{x} . Cases with bad convergence are handled by putting an upper limit on γ (`CLS_GA_UPPER_LIMIT`) and on the number of iterations (`CLS_MAX_ITER`).

In a precursor to Qpack more elaborated iteration schemes were tested [13]. Different versions to find the γ giving the maximum decrease of the cost function for each iteration were tested. Despite that a lower number of iterations could be achieved, the total calculation time was little affected and could even be increased. The reason for this is that the extra calculation burden of going to a new iteration is to determine \mathbf{K}_i , and as this part was rapid in the used forward model [8], it could be more time consuming to test more values for γ (requiring the calculation of $F(\mathbf{x}_i, \mathbf{b})$) than just going to the next iteration. As the calculation of Jacobians in ARTS uses analytical expressions, as in [8], the same behaviour was expected in Qpack and only a simple scheme for the updating of γ was implemented. On the other hand, Qpack contains some options to avoid unnecessary recalculation of absorption and Jacobians (`CLS_RECALC_ABS_ON` and `CLS_RECALC_WFS_NITER`, respectively), with the obvious purpose to speed up the calculations.

ARTS can provide species Jacobians for different units. The option used by Qpack is to provide the Jacobians for a relative change of the state. This corresponds to the retrieval unit used in Qpack. However, for nonlinear inversions and iterations beside the first one, a rescaling of the Jacobians is needed as ARTS is feed with \mathbf{x}_i , and the Jacobians are then valid with respect to that

state, instead of \mathbf{x}_a as required. The needed rescaling is achieved by dividing the Jacobians by their corresponding value in \mathbf{x}_i .

A positive constrain can be applied for species profiles (CLS_SPECIES_POS_ON). This is achieved by inverting the (natural) logarithm of the abundance, instead of the abundance directly. In [8] it is shown that the species Jacobians to apply with a logarithmic transformation are obtained by multiplying the standard Jacobians with their corresponding value in \mathbf{x}_i , a step which happens to perfectly counteract the scaling discussed in the paragraph above. See Section 7 for the definition of the species covariance matrices with the positive constrain selected.

6.3. Inversion characterisation

A general characterisation of an inversion case is obtained by the function `qpcls_invchar`. An example of the output of this function is shown in Fig. 1. The upper left plot shows the measurement response, \mathbf{w} , defined as [14]

$$\mathbf{w}^q(i) = \frac{\mathbf{A}_i^q \mathbf{x}_a^q}{\mathbf{x}_a^q(i)}, \quad (17)$$

where $\mathbf{x}_a^q(i)$ is value i of the part of the a priori vector that corresponds to retrieval quantity q , and \mathbf{A}_i^q is row i of the (square) matrix corresponding to \mathbf{x}_a^q . The measurement response can be used as an indicator on the contribution from the measurement to the retrieved value, compared to a priori information. The measurement response should ideally equal one. For retrieved values with $\mathbf{w}(i) \approx 1$, at least broad scale features of the true state will be mapped to the retrieved state maintaining close to correct magnitude. However, short scale features of $(\mathbf{x} - \mathbf{x}_a)$, compared to the vertical resolution, will be damped even for $\mathbf{w}(i) = 1$, and then add to the smoothing error (see below).

The upper right plot displays the standard deviation of total error retrieval error, defined in Eq. (7), and the observation error,

$$\mathbf{D}_y \mathbf{S}_e \mathbf{D}_y^T. \quad (18)$$

The difference between these two errors is the smoothing error,

$$(\mathbf{A} - \mathbf{I}) \mathbf{S}_x (\mathbf{A} - \mathbf{I})^T, \quad (19)$$

caused by the limited resolution of the retrieval process.

The middle left plot shows the resolution length, measured by the full width at half maximum (FWHM) of the averaging kernels, which are displayed in the middle right plot. The averaging kernels shown correspond to the definition of \mathbf{A}_i^q in Eq. (17), that is, the parts corresponding to other retrieval quantities are omitted.

The lower left plot shows the correlation of the total retrieval error, between values of $\hat{\mathbf{x}}$ belonging to the same retrieval quantity. The error correlation matrix, \mathbf{C}_δ is obtained as

$$\mathbf{C}_\delta(i,j) = \frac{\mathbf{S}_\delta(i,j)}{\sqrt{\mathbf{S}_\delta(i,i)} \sqrt{\mathbf{S}_\delta(j,j)}}, \quad (20)$$

where i and j are row and column indexes.

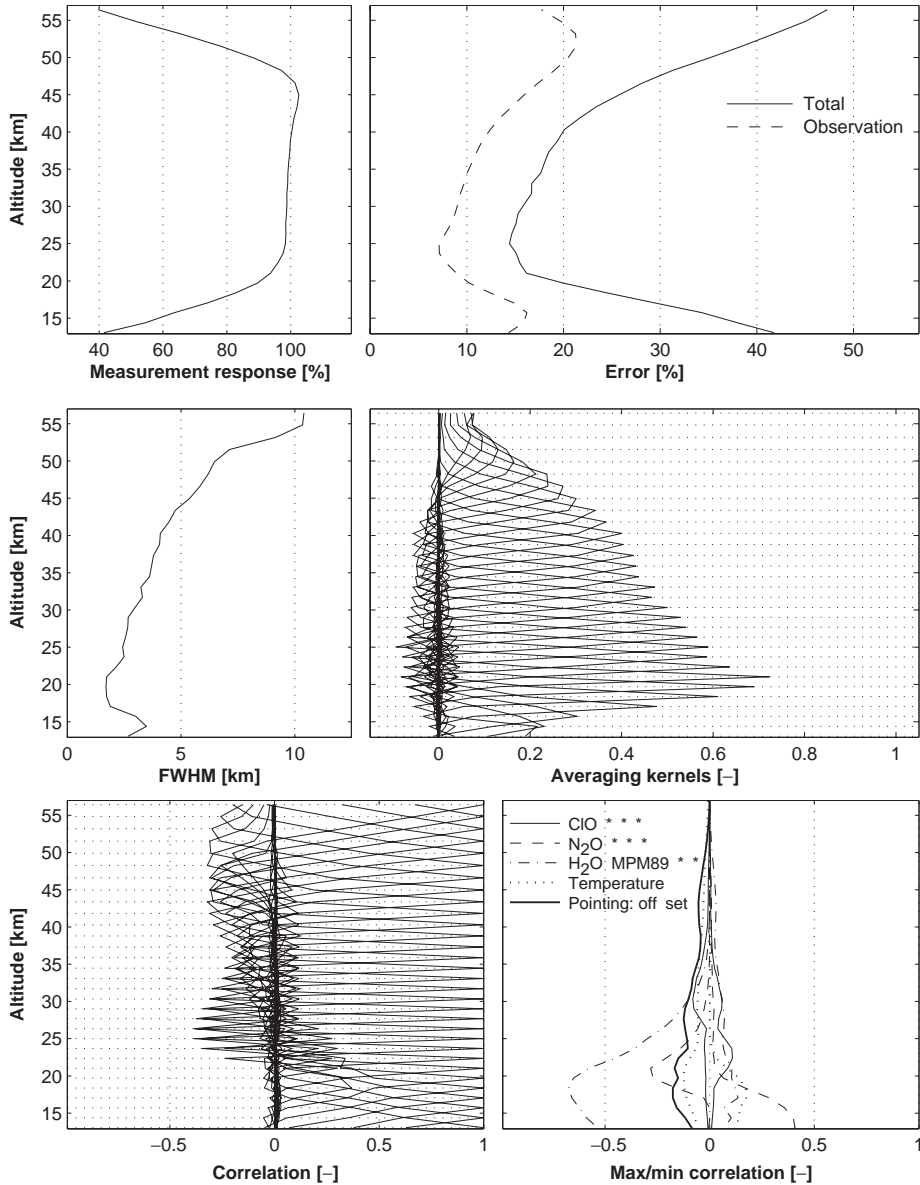


Fig. 1. An example of general inversion characterisation plot created by `qpcls_invchar`. Plots like this are produced if the function is called without output arguments. Otherwise, the function returns the data shown in the figure. The example shows ozone inversion performance for a hypothetical limb sensor measuring emission in the range 501.18–501.58 GHz (see Fig. 4). The plot is described in the text.

The lower right plot gives information on the error correlation between different retrieval quantities. For each position of \hat{x} for the quantity of the plot, the minimum and maximum correlation to the retrieved values for all other retrieval quantities are plotted. For example, if the two quantities are both vertical profiles, the plot shows the minimum and maximum correlation to

all altitudes of the other profile. If the other quantity is a scalar, there is only one correlation value and the minimum and maximum correlation are equal.

The calculations of `qpcls_invchar` do not involve the inversion of a particular measurement. The results are then valid for the a priori state. The actual state should not influence strongly the general inversion characteristics beside for strongly nonlinear situations (see further [1]). However, averaging kernels and estimates on the observation and smoothing errors (including full covariance matrices) can be returned by `qpcls` for the particular inversion performed. As for `qpcls_invchar`, the characterisation considers only quantities with a do-level of 2 or 3.

The retrieval error caused by individual error sources is obtained by the function `qpcls_errorplot` (Fig. 2). This function considers also quantities with do-level 1, and plots the diagonal elements of $\mathbf{D}_y \mathbf{K}_q \mathbf{S}_q \mathbf{K}_q^T \mathbf{D}_y^T$ or $\mathbf{D}_y \mathbf{S}_e \mathbf{D}_y^T$, depending if the error quantity is treated as a forward model parameter or a measurement error, where \mathbf{K}_q and \mathbf{S}_q are the Jacobians and covariance matrix, respectively, for the considered quantity.

It should be noted that all inversion and error characteristics are calculated for the linear version of OEM. That is, the matrix \mathbf{D}_y is calculated following Eq. (8), that corresponds to setting $\gamma = 0$ for nonlinear retrievals. To evaluate the retrieval error with $\gamma > 0$ would give an underestimation.

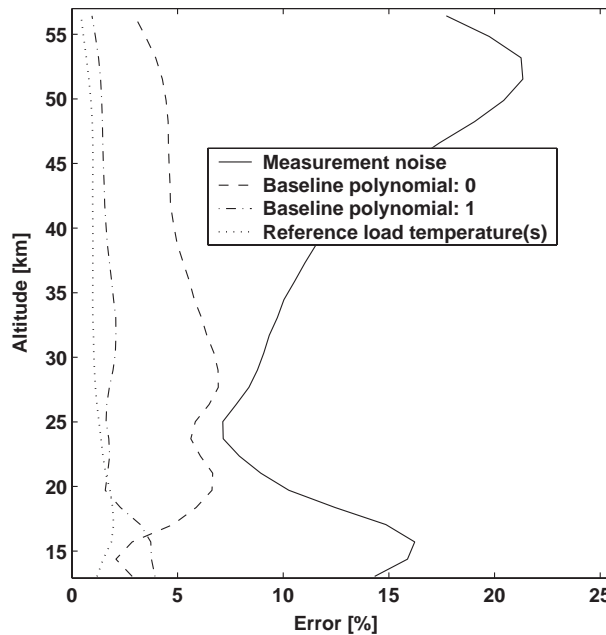


Fig. 2. Example on error characterisation plot created by `qpcls_errorplot`. Plots like this are produced if the function is called without output arguments. Otherwise, the function returns the data shown in the figure. Simulation conditions as for Fig. 1. Measurement thermal noise was set to do-level 2, while the other error sources were set do-level 1. This means that the observation error in Fig. 1 and error due to measurement noise here are identical.

7. Conditional simulations

The term conditional simulations refers here to the generation of a set of atmospheric and sensor states, and their corresponding spectral values conditioned to some given statistical assumptions. A typical use is error characterisation. If the mapping between atmospheric and sensor states and spectra is not linear, the error characterisation suggested in Section 6.3 is not valid. Then conditional simulations can be used to characterise the retrieval error inverting the set of generated spectra and then comparing true and inverted atmospheric states. Another application is to generate regression sets for statistical inversions approaches, for instance, to build a random set to train neural networks.

The random realizations are handled by three function calls: `qp_rnd_atm(Q)`, `qp_rnd_sensor(Q)` and `qp_rnd_atmxsensor(Q)`. The first function generates a set of atmospheric realizations and runs ARTS to produce the corresponding monochromatic pencil beam spectra; the second generates the sensor realizations; the third applies the sensor to the monochromatic pencil beam spectra. So far random realizations are implemented for species, temperature, zenith angles observed by the sensor, thermal noise and spectral baseline offsets. The structure `Q` is used to command the functions. If the corresponding master field has a value different from zero (do-levels ≥ 1), realizations following the statistics given by the corresponding covariance matrices are created. The number of realizations is given in `NUMBER_DO`. An example of generated ozone profile is plotted in Fig. 3 and an example of generated spectra for the Odin-SMR radiometer is given in Fig. 4.

For species normal and log-normal distributions are permitted, controlled by `CLS_SPECIES_POS_ON`. For temperature and thermal noise only normal distribution is allowed, for zenith angles normal and uniform distributions can be used by setting `POINTING_PDF`. The log-normal distribution is advised for the species realizations as it prevents the generation of negative values. A set of different mean states can be used for species and temperature, then the individual set of

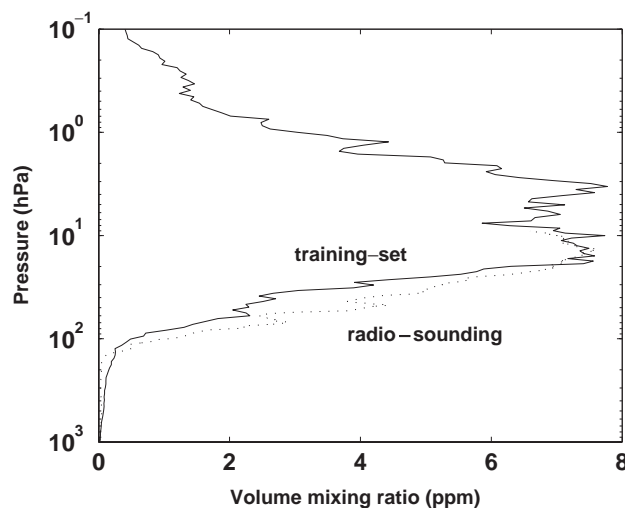


Fig. 3. Example of a randomly generated ozone profile. A profile from radiosonde data is also plotted for comparison.

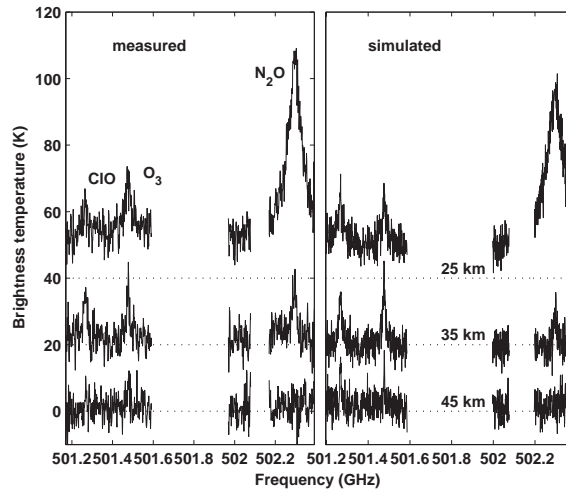


Fig. 4. Example of randomly generated spectra. The spectra correspond to the Odin-SMR radiometer, where three tangent altitudes inside the scan range have been selected. Measured spectra are also plotted for comparisons.

realizations are appended together. This is controlled by APRIORI_VMR_DIR and SETUP_VMR_DIRS.

The normal realizations are generated by applying the Choleski decomposition method [15]

$$\tilde{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{L}\mathbf{r}, \tag{21}$$

where $\tilde{\mathbf{x}}$ is a matrix with random profiles, $\bar{\mathbf{x}}$ is the mean profile, \mathbf{L} is a lower triangular matrix fulfilling $\mathbf{L}\mathbf{L}^T = \mathbf{S}_x$, and \mathbf{r} is a random vector of uncorrelated values, zero mean, unit variance and normal distribution. The log-normal realizations are generated by exponentially transforming the outcomes of a normal distribution. If t is normally distributed as $N(\bar{t}, \sigma_t)$, then the transformation $x = e^t$ results in a log-normal distribution of mean \bar{x} and variance σ_x^2 . The relation between both distributions is

$$\sigma_t^2 = \log\left(1 + \frac{\sigma_x^2}{\bar{x}^2}\right), \tag{22}$$

$$\bar{t} = \log(\bar{x}) - \frac{1}{2} \sigma_t^2. \tag{23}$$

Notice that in this case the covariance matrix defines the normal distribution of t . Care then has to be taken by the user to ensure that the covariance matrix given in the normal space corresponds to the desired statistics in the log-normal space. The relations given in Eqs. (22) and (23) can obviously be used for this. However, for the typical variabilities used to disturbed atmospheric profiles ($\sigma_x < 0.5$) the differences are small and they can be used interchangeably. Note also that this has implications for the retrievals if the same covariance matrix used for the generation of the atmospheric states is used for the retrieval. If a log transformed species retrieval of the log-normal generated states is performed (see Section 6.2), the given \mathbf{S}_x in the normal space t are the right statistics. If \mathbf{S}_x were the statistics in the log-normal space, then \mathbf{S}_x for the retrieval would have not

been the right statistical description. However, as it was said before, the differences in practice are small and for typical variabilities the inversions would have been very close in both cases.

8. Some practical considerations

Qpack contains some functionality to set up input files that are critical for the trade-off between calculation speed and accuracy. The function `qpopt_fmono` selects the grid to be used for monochromatic calculations in ARTS. Inside Qpack, spectra are treated as piecewise linear functions between the monochromatic frequencies points, and `qpopt_fmono` generates a frequency grid for which the maximum deviation between the piecewise linear representation and a very detailed reference calculation is below a selectable limit. The function can perform this operation for an arbitrary number of viewing directions in parallel, to consider the fact that the shape of spectra can vary strongly during a limb scan.

The spectroscopic data for line-by-line calculations in ARTS are taken from a separate file, the linefile. The simplest approach to generate a linefile would be to just copy all transitions inside some frequency range from a spectroscopic database to the linefile, but as the transitions vary strongly in strength it would be more favourable to sort out the transitions giving the highest contribution to the spectra. The function `qpopt_linefile` performs such a selection, by calculating the contribution for all transitions inside the given frequency range and keeping only those exceeding a specified threshold. So far the input to `qpopt_linefile` has to follow the Verdandi data format. Verdandi is a merge of the JPL and HITRAN catalogues, with some data replaced from other sources when judged to be more accurate. Verdandi can be downloaded from www.rss.chalmers.se/gem/Research/verdandi.html

Quantities that must be considered in the retrieval process, but whose actual values are of no interest, can be treated as observation uncertainties. Thermal noise is the standard example on such a quantity, but also other error sources can be treated in the same way. For a linear inversion by OEM (Eq. (8)) exactly the same result is obtained for the main retrieval quantities if a secondary quantity is retrieved or treated as an observation uncertainty. This is the case as OEM has the same physical and statistical information at hand in both cases; it does not matter if the uncertainty for a variable is described by S_x or S_e . The situation is somewhat different for nonlinear retrievals, as the influence of observation uncertainties on the spectra can then vary with the actual state.

Correlation for thermal noise between spectrometer channels or spectra from different viewing directions, or inclusion of quantities beside thermal noise among the observation uncertainties, results in that S_e will have non-zero values outside the diagonal. OEM requires the inversion of S_e (or another matrix of the same size), which can be a very costly operation with off-diagonal elements. The computational cost increases rapidly with the size of S_e and it can then be very important to apply data reduction. The efficient eigenvector reduction supported by Qpack (Section 4) makes it possible to handle most cases with no limitations on data part of S_e .

9. Conclusions

Passive remote sensing is today an indispensable technique for atmospheric monitoring. The indirect nature of the observations has the consequence that obtained spectra cannot be analysed

without the aid of a forward model and a retrieval tool. The same is also true for some active techniques, such as limb radio occultations. The components instrument, forward model and retrieval tool constitute together an observation system. This paper describes a practical implementation of a retrieval environment and the sensor part of a forward model. Sensor modelling and practical aspects around the inversions are probably the two least discussed parts of observation systems around atmospheric instruments. The software of concern is denoted as Qpack. Atmospheric radiative transfer is handled by interfacing to the ARTS software (Buehler et al., this issue). Both ARTS and Qpack are distributed freely following GNU public license.

The implementation of Qpack has shown that it possible to combine a high level of generality, flexibility and calculation speed in a retrieval environment. The generality of the software is proved by the fact that it has already been applied on a variety of applications, both for ground-based and satellite sensors. The problem of handling different observation geometries is mainly a task for the forward model. Qpack is general also in the respect that it handles a relatively high number of retrieval and error quantities. The flexibility is achieved in several ways. For example: main quantities (\mathbf{H} , \mathbf{S}_x and \mathbf{S}_e) can be defined separately; detailed statistical information (in the form of covariance matrices) can easily be specified; quantities can be changed from being a retrieval quantity to be treated as an observation uncertainty (or vice versa) in a consistent manner by modifying only a single setting; and retrievals, analytical error characterisation and conditional simulations are handled inside a single environment, where the same data structure can be used throughout. Competitive calculation speed is obtained primarily by analytical expressions for most Jacobians (part of ARTS), new solutions for sensor modelling and data reduction, and optimisation of critical input files (monochromatic frequencies and molecular transitions). In fact, the calculations were found to be sufficiently rapid to use Qpack/ARTS as the computational core for operational inversions of data coming from the Odin sub-millimetre radiometer.

References

- [1] Rodgers CD. Inverse methods for atmospheric sounding: theory and practise, 1st ed. Singapore: World Scientific Publishing; 2000.
- [2] Forkman P, Eriksson P, Winnberg A, Garcia R, Kinnison D. Longest continuous ground-based measurements of mesospheric CO. *GRL* 2003;30:1532.
- [3] Eriksson P, Jiménez C, Murtagh D, Elgered G, Kuhn T, Bühler S. Measurement of tropospheric/stratospheric transmission at 10–35 GHz for H/sub 210 retrieval in low Earth orbiting satellite links. *Radio Sci* 2003;38:8096.
- [4] Murtagh D, Frisk U, Merino F, Ridal M, Jonsson A, Stegman J, Witt G, Eriksson P, Jiménez C, Megie G, de la Noë J, Ricaud P, Baron P, Pardo JR, Hauchcorne A, Llewellyn EJ, Degenstein DA, Gattinger RL, Lloyd ND, Evans WFJ, McDade IC, Haley C, Sioris C, von Savigny C, Solheim BH, McConnell JC, Strong K, Richardson EH, Leppelmeier GW, Kyril E, Auvinen H, Oikarinen L. An overview of the Odin atmospheric mission. *Canad J Phys* 2002;80:309–19.
- [5] Jiménez C, Eriksson P, Murtagh D. Inversion of Odin limb sounding sub-millimeter observations by a neural network technique. *Radio Sci* 2003;38:8062.
- [6] Urban J, Baron P, Lautié N, Schneider N, Dassas K, Ricaud P, Noë Jdela. Moliere (v5): a versatile forward- and inversion model for the millimeter and sub-millimeter wavelength range. *JQSRT* 2004;83:529–54.
- [7] Rodgers CD. Characterization and error analysis of profiles retrieved from remote sounding measurements. *J Geophys Res* 1990;95:5587–95.

- [8] Eriksson P, Merino F, Murtagh D, Baron P, Ricaud P, de la Noë J. Studies for the Odin sub-millimetre radiometer: 1. Radiative transfer and instrument simulation. *Canad J Phys* 2002;80:321–40.
- [9] Eriksson P. Analysis and comparison of two linear regularization methods for passive atmospheric observations. *J Geophys Res* 2000;105(D14):18157–67.
- [10] Rodgers CD. Retrieval of atmospheric temperature and composition from remote measurements of thermal radiation. *R Geophys Space Phys* 1976;14:609–24.
- [11] Eriksson P, Jiménez C, Bühler S, Murtagh D. A Hotelling transformation approach for rapid inversion of atmospheric spectra. *JQSRT* 2002;73:529–43.
- [12] Eriksson P, Chen D. Statistical parameters derived from ozonesonde data of importance for passive remote sensing of ozone. *Internat J Remote Sensing* 2002;23:4945–63.
- [13] Eriksson P, Jiménez C. Non-linear profile retrievals for observations in the lower stratosphere with the odin sub-mm radiometer. In: *IGARSS'98, Sensing and Managing the Environment*, IEEE Publ. 98CH36174, vol. 3. New York: Inst. Electr. Electron. Eng., 1998. p. 1420–3.
- [14] Baron P, Ricaud P, de la Noë J, Eriksson P, Merino F, Murtagh D. Studies for the Odin sub-millimetre radiometer: 2. Retrieval methodology. *Canad J Phys* 2002;80:341–56.
- [15] Cressie N. *Statistics for spatial data*. New York: Wiley Interscience; 1993.