Inverse Problems **26** (2010) 095003 (15pp) doi:10.1088/0266-5611/26/9/095003

An inverse radiative transfer model of the vegetation canopy based on automatic differentiation

M Voßbeck[1], M Clerici[2], T Kaminski[1], T Lavergne[3], B Pinty[4,5] and R Giering[1]

[1] FastOpt GmbH, Schanzenstrasse 36, D-20357 Hamburg, Germany
[2] European Commission, DG Joint Research Centre, Institute for Environment and Sustainability, Global Environment Monitoring Unit, TP 440, via Enrico Fermi 2749, I-21027 Ispra (VA), Italy
[3] Norwegian Meteorological Institute, PO box 43, Blindern, N-0313 Oslo, Norway
[4] European Commission, DG Joint Research Centre, Institute for Environment and Sustainability, Global Environment Monitoring Unit, TP 272, via Enrico Fermi 2749, I-21027 Ispra (VA), Italy
[5] Seconded to the Earth Observation Directorate, ESA-ESRIN, via Galileo Galilei, Casella Postale 64, 00044 Frascati, Italy

E-mail: Michael.Vossbeck@FastOpt.com, marco.clerici@jrc.ec.europa.eu, Thomas.Kaminski@FastOpt.com, thomasl@met.no, bernard.pinty@jrc.ec.europa.eu and Ralf.Giering@FastOpt.com

Received 18 November 2009, in final form 11 May 2010
Published 9 July 2010
Online at stacks.iop.org/IP/26/095003

Abstract
This paper presents an inverse model of radiation transfer processes occurring in the solar domain in vegetation plant canopies. It uses a gradient method to minimize the misfit between model simulation and observed radiant fluxes plus the deviation from prior information on the unknown model parameters. The second derivative of the misfit approximates uncertainty ranges for the estimated model parameters. In a second step, uncertainties are propagated from parameters to simulated radiant fluxes via the model's first derivative. All derivative information is provided by a highly efficient code generated via automatic differentiation of the radiative transfer code. The paper further derives and evaluates an approach for avoiding secondary minima of the misfit. The approach exploits the smooth dependence of the solution on the observations, and relies on a database of solutions for a discretized version of the observation space.

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Atmospheric carbon dioxide is the single most important anthropogenic greenhouse gas. Its continued increase is the major reason for observed global warming [1]. The increase is clearly of anthropogenic origin, but it is tempered by uptake from natural reservoirs. Despite considerable efforts in monitoring the global carbon cycle, the magnitude and

 © 2010 IOP Publishing Ltd Printed in the UK & the USA

distribution of carbon dioxide sources and sinks are still uncertain. One of the reasons relates to a traditional observing system, which relies on sparse network sampling, e.g. atmospheric concentrations [2], or exchange fluxes with the ocean [3] or the terrestrial biosphere (see http://www.fluxnet.ornl.gov). Earth observation strategies from space are able to provide information at medium spatial resolutions (typically of the order of 1 km) over the entire globe on a regular basis, e.g. the moderate resolution imaging spectroradiometer (MODIS), multiangle imaging spectro radiometer (MISR), sea-viewing wide field-of-view sensor (SeaWiFS), global imager (GLI), medium resolution imaging spectrometer (MERIS) or VEGETATION.

Remote sensing of the terrestrial vegetation poses a classical inverse problem. The radiation signal observed by a remote sensing instrument at the top-of-atmosphere is analysed in order to first remove at best the contamination by atmospheric constituents and second to generate albedo products (hereafter denoted as BiHemispherical Reflectances, BHRs) characterizing the scattering probability of the coupled vegetation soil systems. This same albedo can be simulated by radiative transfer (RT) models that solve the radiative balance within the canopy and the background, by modelling the physical and optical properties of the vegetation and the underlying surface. The model characterizes the vegetation soil system via a set of parameters. The inverse problem consists in the estimation of these parameters from the retrieved albedos. This inverse problem is usually under-determined, i.e. the number of (unknown) parameters exceeds the number of observations.

One of the traditional approaches to the inverse problem is via so-called look up tables (LUTs) [4]: in a preliminary step, the parameter space is sampled over a discrete grid, the RT model runs at each sample point and the simulated observation counterparts are recorded in an LUT. For any given set of observations, the inversion procedure consists in a search of the LUT for the simulated counterpart that is closest to the observations, and the solution of the inverse problem is the parameter set it was simulated from. The LUT approach suffers from a number of disadvantages: first, the grid in parameter space must be coarse, in order to limit both the number of runs of the RT model and the time spent for searching the LUT. Note that computational efficiency is essential, because the processing of global or continental scale observational data sets requires millions to billions of inverse model runs.

Second, due to the under-determined nature of the inverse problem, the best fit to the observations is typically achieved on an entire manifold in parameter space. In this situation, the solution provided by the LUT procedure is unstable. For instance, it is highly sensitive to the specification of the grid in parameter space: a slight change of the grid may shift the solution to a completely different region in parameter space. Third, it is difficult to furnish the solution with uncertainty ranges (error bars) that are consistent with uncertainties in the observations and the RT model. These uncertainty ranges are essential for the use of the estimated parameter values to better constrain models representing the carbon, water and energy cycle.

This paper presents an alternative approach, which regularizes the inverse problem via *a priori* information on the parameters. In contrast to the LUT approach, the parameter space is searched by a gradient algorithm for a minimum of the misfit to the observations, including a term that quantifies the deviation from the prior parameter values. The standard starting point for this iterative search is the prior parameter set. Uncertainty estimates for the optimal parameters are approximated via the inverse of the second derivative of this misfit at its minimum. The necessary derivative information is provided via automatic differentiation (AD) [5]. The AD tool TAC++ [6] is used to transform the C++ source code of the RT model into a highly efficient code for the evaluation of first (adjoint code) and second (Hessian code) derivatives. Our inverse modelling system, the Joint Research Centre Two-stream

Inversion Package (JRC-TIP), is built around the RT code [7] which provides a two-stream type of solution to the radiation transfer problem in vegetation canopies. The system has been successfully applied to estimate seven vegetation parameters over a number of sites [8, 9] from observed BHRs in two spectral broadbands, namely the visible (0.3–0.7 μm) and near-infrared (0.7–3.0 μm) domains, i.e. there are two observations to constrain seven parameters. The challenging objective is to apply JRC-TIP in the same configuration but to observational data sets that provide a synoptic coverage of continents or the entire globe. An example is described by M Clerici *et al*[10], who process an observational data set over Europe provided by the MODIS sensor in 2005. Given a pixel size of 1 km, the application requires on average 12.0 million inverse problems to be solved for every available 16 day period over a year. The present paper describes a set of extensions of JRC-TIP, tailored to enhance robustness and computational efficiency for such large-scale applications.

The remainder of this paper is organized as follows: section 2 describes the model and the observational data. Section 3 introduces JRC-TIP. Section 4 describes and demonstrates the extended package. Finally, section 5 gives a summary and conclusions.

2. The radiation transfer model

Our inversion package, JRC-TIP, is built around the two-stream radiation transfer scheme developed by Pinty *et al* [7]. The scheme calculates the radiation transfer within the vegetation canopy. It is designed for use in climate models, which, driven by the need of computational efficiency, can only accommodate one-dimensional schemes. In this approach, three-dimensional effects are taken into account via so-called effective state variables that characterize the radiative properties of the vegetation layer and the albedo of the background that corresponds to the 'true' (by contrast to effective) value. In total there are four state variables for each spectral band. Three of them, the leaf reflectance r_l, transmittance t_l (or alternatively the leaf single scattering albedo $\omega_l = r_l + t_l$ and asymmetry factor $d_l = r_l/t_l$) and the background reflectance, are wavelength dependent. The fourth one, the leaf area index (LAI), quantifies the density of leaves in the vegetation layer and does not exhibit any spectral dependence. The scheme calculates the fluxes scattered by (albedo, R), absorbed in (A) and transmitted through (T) the vegetation canopy. The fluxes are normalized by the incoming solar flux, i.e all three fluxes take values in the interval [0, 1]. The appendix details the functional dependence of the simulated fluxes on the state variables (in the following denoted parameters) via the model equations. Results simulated by the two-stream model have been compared against those delivered by a Monte Carlo model under a large variety of standard and unfavourable canopy conditions [7]. The differences between both simulations were found to be confined within the 3% relative accuracy range of the scattered and transmitted fluxes. Further, the model participated in the radiation transfer model inter-comparison (RAMI) initiative for benchmarking models of the radiative transport on the terrestrial surface (http://rami-benchmark.jrc.ec.europa.eu). Figure 1 illustrates the flow of information from parameters to radiant fluxes for the case of two wavelengths. It is convenient to introduce a compact notation, with $\tilde{x}$ denoting the vector of all parameters, $\tilde{y}$ denoting the vector of all radiant fluxes and

$$\begin{array}{rccc} M : & \mathbb{R}^7 & \to & \mathbb{R}^6 \\ & \tilde{x} & \mapsto & \tilde{y} \end{array}$$

denoting the model. In principle, all of the radiant fluxes are observables, but typical space applications such as the large-scale application in section 4 use observations of the BHRs in two broad visible and near-infrared spectral bands. Denoting by P the projector onto the

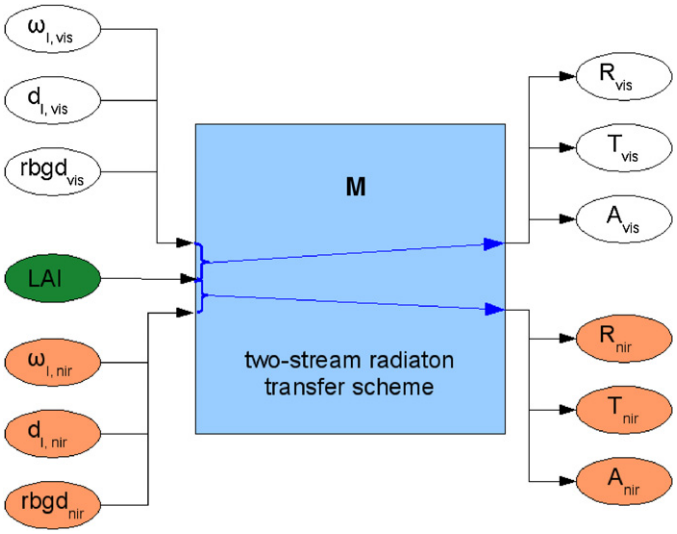

Figure 1. Flow of information from parameters to radiant fluxes. Indices vis and nir respectively denote visible (0.3–0.7 μm) or near-infrared (0.7–3.0 μm) versions of process parameter or radiant fluxes.

observed subspace of radiant fluxes, we can define a mapping $M_P = P \circ M$ from the parameter space into the observed sub-space.

3. Inverse model

Our inverse model, the JRC-TIP, delivers a set of variable values entering the two-stream forward model from a given set of observations and the available prior information. Our methodological approach is based on [11]: the *a priori* state of information is quantified by a probability density function (PDF) in parameter space, the observational information by a PDF in observation space and the information from the model by a PDF in the joint space, i.e. the Cartesian product of parameter and observation spaces. The inversion combines all three sources of information and yields a posterior PDF in the joint space.

Prior and observational PDFs are difficult to specify. We use Gaussian shapes with respective mean values denoted by x_0 and d and respective covariance matrices denoted by $C(x_0)$ and $C(d)$; the square roots of the seven diagonal entries of the prior covariance matrix are respectively denoted by $\sigma_i(x_0)$.

Since the model is only weakly nonlinear, we can approximate the posterior PDF by a Gaussian PDF. The corresponding marginal PDF in parameter space is thus also approximately Gaussian, with the mean value x and covariance $C(x)^{-1}$. The mean x is approximated by the maximum likelihood point, i.e. the minimum of the misfit function, $J : \mathbb{R}^7 \to \mathbb{R}$:

$$J(\tilde{x}) = \tfrac{1}{2}[(M_P(\tilde{x}) - d)^T C(d)^{-1}(M_P(\tilde{x}) - d) + (\tilde{x} - x_0)^T C(x_0)^{-1}(\tilde{x} - x_0)]. \quad (1)$$

$C(x)$ is approximated by the inverse of the misfit function's Hessian, H, evaluated at x (see also [11]):

$$C(x) = H(x)^{-1}. \quad (2)$$

To understand this relation it is useful to look at the case of a linear model (denoted by M'_p):

$$H(x) = {M'_p}^T C(d)^{-1} M'_p + C(x_0)^{-1}. \tag{3}$$

The Hessian is the sum of two terms: one reflecting the strength of the constraint by the prior information, and the other reflecting the observational constraint. Typically adding the observational constraint increases the curvature of the cost function which via equation (2) translates to a reduction in uncertainty compared to the prior.

From the optimal parameter set we can simulate all radiant fluxes (including the non-observed ones) via $y = M(x)$. To assess the strength of the observational constraint on these simulated radiant fluxes, we use M', the first derivative of M to propagate posterior parameter uncertainties forward to the radiant fluxes:

$$C(y) = M' C(x) {M'}^T. \tag{4}$$

This diagnostic step is particularly useful for comparing the TIP results with independent observations.

Equation (1) is implemented around the C software code of the original two-stream version. The minimization of J uses a quasi-Newton algorithm with BFGS updates [12], which relies on the code for evaluating J and on the code for evaluating its gradient.

The code for evaluating the gradient is generated by AD of the function code: the function code is decomposed into elementary functions (such as $+$, $-$, $\sin(\cdot)$), which (more or less) correspond to the individual statements in the code. Differentiating these elementary functions is simple and yields the so-called local Jacobians. According to the chain rule, the product of the local Jacobians yields the derivative of the composite function. There are two main strategies for evaluating this multiple matrix product. The tangent code uses the order in which the original function code evaluates the statements to evaluate the product of their local Jacobians. The adjoint code performs this evaluation in the reverse order. In AD terminology, the tangent code operates in forward mode, and the adjoint code operates in reverse mode. Similar to the finite difference approximation, the computational resources needed in forward mode increase with the number of independent variables. In reverse mode, they are roughly proportional to the number of dependent variables and virtually do not depend on the number of independent variables. In the nonlinear case the local Jacobians depend on the point, where they must be evaluated. Within the function code this point is associated with a set of variables, the so-called required variables in AD terminology [13]. Since the adjoint code evaluates the local Jacobians in the reverse order of the original function, providing these required variables is more complex and computationally more expensive than in the tangent code. Hence, the reverse mode is only favourable when the number of independent variables exceeds the number of dependent variables by a break even value, which, for a scalar-valued function, is typically in the range of 2–5 independent variables. As opposed to derivative approximation by finite differences (also known as numerical differentiation), AD provides sensitivity information that has the same accuracy as the function code, i.e. it is accurate within the machine's round-off error.

The code of J, which, excluding comments, comprises about 300 lines of C, was differentiated by the AD tool TAC++ [6]. TAC++ implements a number of concepts for practical derivative coding suggested in [14]. This implementation is based on the same concepts as the AD tool transformation of algorithms in Fortran (TAF) [13], which is being successfully applied for almost a decade to an ever increasing number of Fortran 77-95 codes from a wide range of disciplines [15]. This includes large-scale applications to codes of up to 300 000 lines excluding comments.

TAC++ can generate both tangent and adjoint codes. With seven independent variables and one dependent variable the adjoint code is computationally more efficient. Depending on

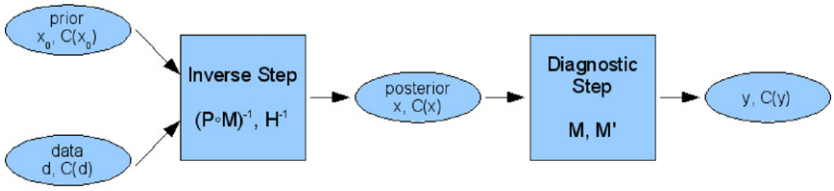

Figure 2. Schematic flow of information through JRC-TIP.

platform, compiler and compiler options, it takes the CPU time of 3–4 function evaluations to evaluate both the function and its gradient. This is the typical range for TAC++ applications; for further examples we refer to [6].

The code for evaluating the Hessian in equation (2) is generated by reapplying TAC++ in forward mode to the adjoint code (forward over reverse mode). This code does evaluate the entire Hessian (and the gradient) in one run, i.e. it simultaneously evaluates seven directional derivatives of the gradient of J. This takes a CPU time comparable to about 23 evaluations of J. The inversion follows the computation of Hessian's eigenvalues via the cyclic Jacobi method [16].

The code for evaluating the Jacobian M' is generated by applying TAC++ in forward mode to the code of M. This code evaluates M and M' in one run and takes a CPU time comparable to about 2.2 evaluations of M.

The flow of information through JRC-TIP is summarized in figure 2. The inverse step consists of the minimization of J, and the evaluation and inversion of $H(x)$. The iterative minimization procedure starts from the prior value x_0. The diagnostic step evaluates M and propagates the posterior uncertainties from parameters to radiant fluxes.

The inversion package has first been applied over single sites with known characteristics to estimate parameters and radiant fluxes, together with their uncertainties, from visible/near-infrared BHR pairs provided by MISR or MODIS sensors [8, 9]. Note that although two-dimensional observational data sets are typical for many applications, the package is capable of handling any combination of radiant fluxes as observables. A typical run of the inversion step requires about 1 ms on a Linux platform with Intel Xeon 2.5 GHz CPU, gcc version 4.3.2 compiler and '-O2 -msse2 -ffast-math -funroll-loops' options (platform 1). Another Linux platform with two Intel 3.16 GHz CPUs, gcc version 4.3.3 compiler and '-O3 -ffast-math -funroll-loops -msse4.1 -static' options (platform 2) achieves about twice that speed.

4. Extending the inverse model

This section discusses a series of procedures aiming at increasing the performances and in particular the robustness of the JRC-TIP. This has been motivated by the methodological challenges arising from continental-scale or global-scale applications of the package. For an example (see [10]) we use BHR observations derived by the MODIS sensor in the visible and near-infrared spectral bands as available from MODIS collection 5 products (MCD43B3) generated at 0.01 degree resolution for successive 8 and 16 day periods. We use the level 3 MODIS product (MCD43B3) provided every 16 days by the sensor on-board Terra platform (so-called collection 5). $C(d)$ is assumed to have only zero off-diagonal elements; for the diagonal we use the square of a σ that corresponds to 5% of the observed value it refers to. σ values below 0.0025 (corresponding to a BHR value below 0.05) are, however, set to the minimum value (floor) of 0.0025. Given a regular latitude/longitude grid of 0.01 degree (around 1 km), the extended European window shown in figure 7 covers 22.5 million pixels.

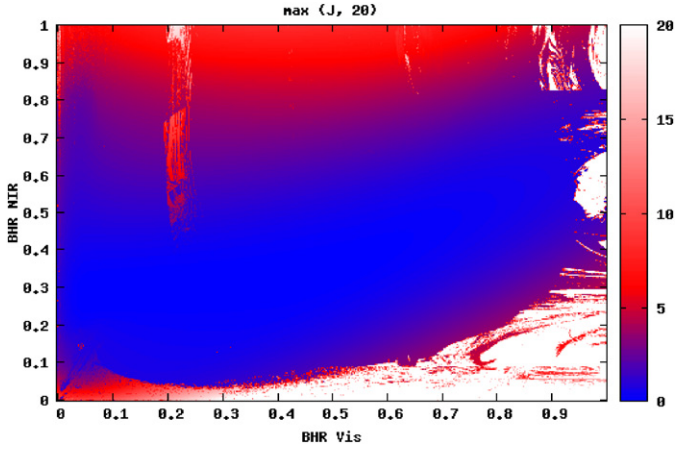

Figure 3. Distribution of posterior J over the observation space as extracted from the base version of the TIP table ($C(x_0)$ based on 'snow' case prior information).

Due to the sea extent and missing data, on average about 12 million of those pixels represent valid inputs for the inverse problem.

We are using two sets of prior information: one under normal conditions, and one in the presence of snow. Snow is detected via another satellite product, namely MCD43B2. Both sets are taken from [9], where they are called 'standard' (i.e. not 'green') and 'snow'. In both cases all off-diagonal entries of $C(x_0)$ are zero, except for those among the background reflectance in the two wave bands.

Recall that both observed BHR values are limited to the interval $[0, 1[$. We can cover this interval by a discrete set of 1000 BHR values using a regular grid with a spacing of 0.001, a value considerably smaller than the observational uncertainty that ranges from 0.0025 to 0.05. This provides a discretization of the set of possible observations that includes only 1 million observation pairs. We ran JRC-TIP for each pair in the set and recorded the result in a database, which we denote as the TIP table. Note that the TIP table differs from the tables to be constructed for the LUT approach, because it is based on a discretization of the observation space rather than a discretization of the parameter space.

The TIP table is useful in a number of ways. Our first use is as a test bed for the robustness of JRC-TIP. One possible visualization of the TIP table is by plotting the value of the misfit function for the posterior parameter ($J(x)$) over the observation space, which is shown for the 'snow' case prior parameters (figure 3). This yields a smooth ground surface with a number of outliers above the ground surface. In fact many of the outliers do sit on a second surface above the ground surface (not recognizable in this two-dimensional form of the graph). The concentration of outliers is particularly high for low BHR values in the near-infrared domain. $J(x)$ expresses the degree of consistency between the observations, the prior information and the model. For seven unknowns one would expect a value around 3.5 [11]. While lower values indicate a too conservative choice of uncertainties, higher values indicate too tight uncertainties. Figure 4 displays the near-infrared component of the residual $(M_P(\tilde{x}) - d)$ over the observation space, again based on 'snow' case prior parameters. The residuals confirm the same outliers as the misfit function. Evaluating the forward model M_p at the prior value, $\tilde{x} = x_0$, yields BHR values of 0.29 in the near-infrared and 0.09 in the visible domain. Consequently this pair is associated with a zero residual. Increasing the near-infrared component in the observation produces a negative near-infrared residual, because the

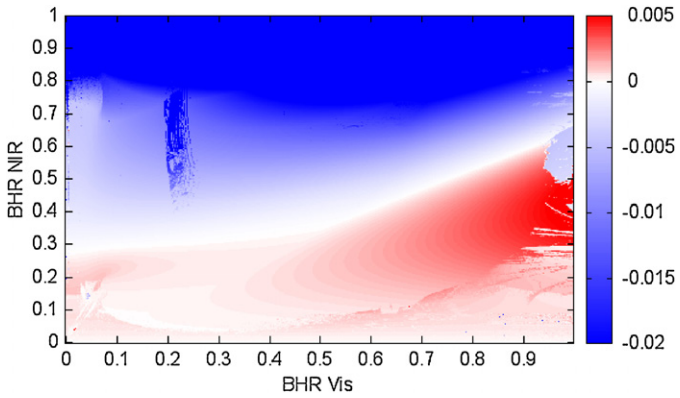

Figure 4. Distribution of the near-infrared component of the residual $(M_P(\tilde{x}) - d)$ over the observation space as extracted from the base version of TIP table ($C(x_0)$ based on 'snow' case prior information).

inversion achieves a compromise between prior and observational terms in the misfit function (equation (1)). Conversely, a reduced near-infrared component produces a positive residual. This effect is modulated by the visible component in the observation, because both wave bands are coupled via the single spectrally independent parameter, LAI.

The occurrence of the outliers is typically caused by the multiple minima of equation (1). For an example figure 5 displays J over a section in parameter space between x_1, the estimated parameter value for a point P_1 in observation space, and x_2, the estimated parameter value for P_2, one of the neighbours with lower minimum. The figure demonstrates that for P_1 the misfit J has a second local minimum close to x_2. In fact this point yields a better fit to both data and prior (and thus a lower value of J), but the minimization algorithm failed to detect it, because it got trapped at x_1. Standard gradient procedures are well known for this behaviour. Finding global minima is in general a challenging task. Typical strategies include evolutionary techniques [17], or Monte Carlo (i.e. random) sampling of the posterior PDF, e.g. the metropolis algorithm [18] or simulated annealing [19]. In the following we will pursue alternative approaches to this problem that fit well within our gradient-based framework.

The path of the iterative minimization procedure through the parameter space depends on the starting point. In the above example, we use the prior value x_0 as the starting point. Starting a new minimization from a different point, we could hope to end up in the minimum near x_2. For the extension of our inversion package we thus need to define a sequence of starting points providing a reasonable sample of the parameter space. Building on the prior value and the prior uncertainties, we define five starting points by

$$\begin{aligned} s_{1,i} &= x_{0,i} \\ s_{2,i} &= x_{0,i} + \sigma_i(x_0) \\ s_{3,i} &= x_{0,i} - \sigma_i(x_0) \\ s_{4,i} &= x_{0,i} + (-1)^i * \sigma_i(x_0) \\ s_{5,i} &= x_{0,i} - (-1)^i * \sigma_i(x_0), \end{aligned}$$

where $i \in \{1, \ldots, 7\}$.

Based on this sequence of starting points we devise a multiple starting point strategy (MSP) that extends the base version of the package, which used the starting point s_1 to build the initial TIP table. The strategy MSP runs inversions from the remaining four starting points

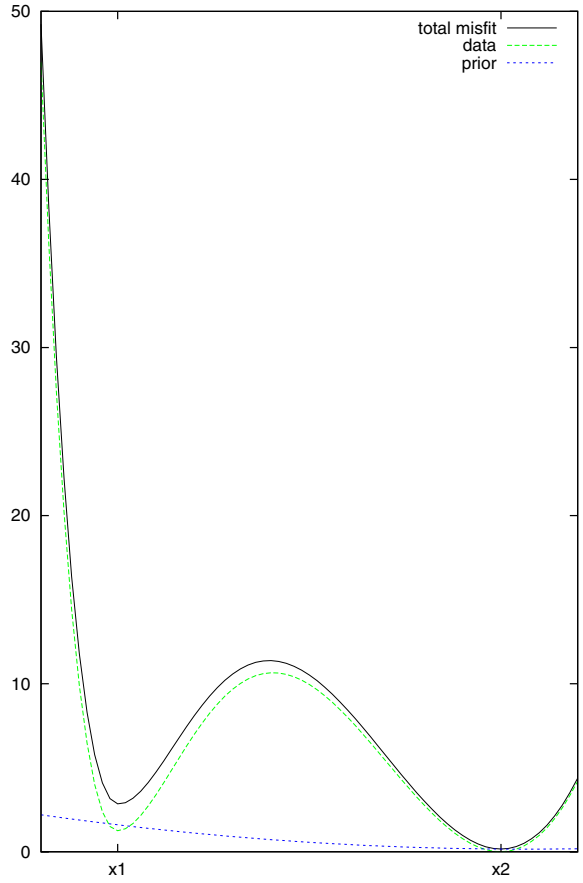

Figure 5. J over a section in parameter space (solid line). Contributions by the prior term (dotted line) and by the data term (dashed line).

of the sequence. Whenever one of these inversions yields a lower value of J, the corresponding entry in the TIP table is updated. A second strategy, multiple starting point threshold (MSPT), proceeds likewise but, to save CPU time, stops the sequence of inversions whenever a J value below a certain threshold (we use a value of 3.0) is reached.

A further strategy to enhance robustness is based on the observation that the misfit and the optimal parameters are smooth functions of d. We define a function $g : \mathbb{R}^7 \times [0, 1]^2 \to \mathbb{R}^7$ via

$$g_j(x, d) = \frac{\partial \tilde{J}}{\partial x_j}(x, d) \qquad (j \in \{1, \ldots, 7\}), \tag{5}$$

where we use $\tilde{J}$ to denote the misfit of equation (1) as a function of both arguments. g is a smooth function with respect to its first and second arguments. For the latter this is obvious from equation (1), whereas regarding the former, g is indeed composed of smooth elementary functions defined by the individual statements in the two-stream code as long as we stay away from denominators of zero. In this respect the code contains two critical spots. At one spot the original code contained a test for a denominator of zero during the function evaluation, which would have terminated the evaluation. At the other spot a square root in the

```
/* ssqrt: "differentiable continuation" for sqrt through "0":

   ssqrt(x,eps) =  sqrt(x),                    x >= eps
                =  x/(2*sqrt(eps)) + sqrt(eps)/2, x < eps
*/
if(ksquare >= LEPSILON) {
   k = sqrt(ksquare);
}
else {
   k = ksquare/(2 * sqrt(LEPSILON)) + 0.5 * sqrt(LEPSILON);
}
```

Figure 6. Alternative code avoiding singular derivative of the square root.

function evaluation opens the possibility of a zero denominator in the corresponding derivative statement. Since this spot only became problematic once derivatives were evaluated, there was no special treatment in the original function code. Hence, we had to add a test for values close to zero and filled this gap in continuously differentiable form as illustrated by the code fragment in figure 6. At `ksquare = LEPSILON` the code is once continuously differentiable but not twice. Thus, there may be points at which g is not differentiable with respect to its first argument, i.e. g is continuous and piecewise continuously differentiable. In the current setup, with two observed BHRs and the above-described prior information we do, however, stay away from this singularity. The singularity can only be approached in cases, where the line search of the minimization algorithm suggests parameter sets far outside the physically meaningful domain. In summary g is continuously differentiable in a domain that includes any reasonable pair of parameters and observations.

Now we select an observation pair d_b and denote by x_b the corresponding minimum of the misfit provided by JRC-TIP. Since $g(x_b, d_b) = 0$ and $\frac{\partial g}{\partial x}(x_b, d_b)$ is the non-singular Hessian of J, by virtue of the implicit function theorem there are a domain $\mathcal{D}$ around d_b and a function $x(d)$ with $x_b = x(d_b)$ and

$$0 = g(x(d), d) = \frac{\partial \tilde{J}}{\partial x}(x(d), d) \qquad (d \in \mathcal{D}). \tag{6}$$

Here $x(d)$ remains a minimum of J since $H(x)$ remains positive definite in $\mathcal{D}$. Moreover, the theorem provides the derivative of $x(d)$:

$$\frac{\mathrm{d}x}{\mathrm{d}d}(d) = -\frac{\partial g}{\partial x}(x(d), d)^{-1}\frac{\partial g}{\partial d}(x(d), d) = -H(x(d))^{-1}M_P'(x(d))^T \qquad (d \in \mathcal{D}), \tag{7}$$

where $(\,)^T$ denotes the transposed. The derivative approximates the sensitivity of the minimum with respect to a small change in the observations. Particularly high sensitivities occur when the orthogonal complement of the null space of M_P' (the row space) projects well onto a small eigenvector of H. Since we are working from the two-stream code rather than from an analytical expression of M_P, we cannot further simplify this expression without ignoring the second derivative of M_P.

In essence the above analysis states that both x and $J(x)$ are smooth functions of d. We can exploit this to devise a further strategy for detecting local minima. The next neighbour starting point (NSP) strategy scans the initial TIP table for points in observation space that yield outliers in the solution. To detect these outliers the misfit at the optimum, the posterior parameter vector or the posterior uncertainties can be used. Our implementation works as follows: in a first step the TIP table is scanned for the local maxima of the misfit of the optimum. All these points are flagged bad, and for them we rerun TIP starting from the posterior parameter vector of the good neighbour with lowest $J(x)$. In case this results in a lower $J(x)$ we update the corresponding entry in the TIP table. We do a few iterations of this

Table 1. Performance of various inversion strategies.

Strategy	NSP iterations	Time (s)	Mean $J(x)$	Max $J(x)$	Extrema	Unrealistic
Base	0	429	4.9801	8572.86	12 191	104 982
	1		4.8403	1710.97	9335	105 124
	14		4.7523	574.28	2168	106 386
MSP	0	2840	4.1441	1181.98	7589	89 893
	1		4.1046	304.49	5832	89 357
	10		4.0799	42.22	2239	89 206
MSPT	0	1280	4.1496	1181.98	8221	92 775
	1		4.1100	304.49	6387	92 241
	9		4.0853	42.22	2328	92 063

procedure, each one resulting in an update of the TIP table, until we cannot further reduce the number of bad points. In a second step we repeat the same procedure, now scanning the TIP table for local extrema of the LAI instead of $J(x)$.

Table 1 evaluates the TIP tables produced with the different strategies. Rows 1–3 refer to the base table without (first row) and combined with the NSP strategy (rows 2 and 3). Row 2 evaluates the TIP table after the first iteration and row 3 after convergence, which in that case was achieved after 14 iterations. Similarly rows 4–6 evaluate the tables produced by MSP and rows 7–9 those produced by MSPT. The third column shows the CPU time required to build the various tables on platform 2. The additional time for the NSP approach is negligible, because the number of extrema (column 6) is very small compared to the observation space. Surprisingly, MSP is about a factor of 7 slower than base, although it performs only five times more inversions. This is because, on average, the inversions starting from perturbed prior values require more iterations to converge. MSPT is about a factor of 2 faster than MSP. We use the mean $J(x)$, maximum $J(x)$, number of extrema and unrealistic values (columns 4 to 7) to quantify robustness. Extrema are a proxy for outliers. Unrealistic values are characterized by at least one parameter value outside the physically reasonable range. We deliberately chose not to exclude the occurrence of such values by the design of our inversion algorithm, because they are useful to diagnose inconsistencies in the observations, the prior information and the model. Without the NSP, the MSP performs better than MSPT, which in turn is superior to base. The NSP reduces the number of outliers considerably, for the base table by about a factor of 6, and for the MSP and MSPT tables by about a factor of 3. The numbers after the first NSP iteration demonstrate that it is useful to carry out more than one iteration. Interestingly, the NSP increases the number of unrealistic solutions when operating on the TIP table generated by the base approach but not when operating on those generated by the MSP and MSPT approaches. Note that typical applications (e.g. [10]) are restricted to a subset of the observation space, in which extrema and unrealistic solutions are less frequent and where the ground surface of figure 3 exhibits moderate $J(x)$.

To illustrate the benefit of the TIP table, the above-mentioned large-scale application has been reprocessed with a revised TIP table that was updated by combining MSPT with the NSP. As compared to the TIP table's base version, this reduces the percentage of unrealistic solutions from 1.43 to 0.21 and the percentage of solutions with $J(x) > 3$ from 6.43 to 0.18. For each observation pair in the data set, the closest pair on the grid is determined, the TIP solution read in and then processed. The form of the processing depends on the application. For instance, it can be the extraction of one of the parameters, e.g. LAI, that is written to

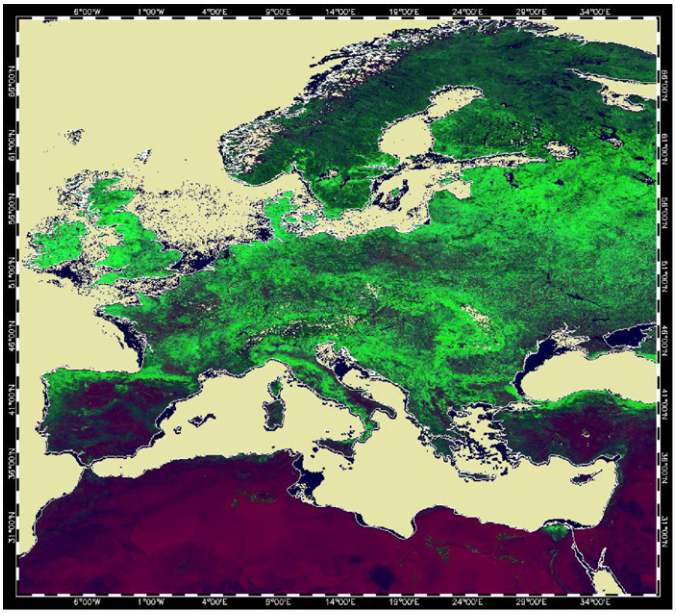

Figure 7. JRC-TIP result over Europe based on the final TIP table.

a file, together with an error bar (reflecting the ± 1 sigma range). It can also be one of the non-observed radiant fluxes, again with the error bar. Or a combination of the parameters can be mapped onto channels for red, green and blue colours to produce an instructive map. As an example, we select the data set from the 16 day period ranging from Julian day 177 to 193 of year 2005, to produce the map in figure 7. The green channel shows LAI, the red channel shows the background reflectivity in the near-infrared domain and the blue channel shows the leaf single scattering albedo ω_l in the visible domain. Whenever there is snow, we simply use white colour. Pixels without observations are black. The remaining bad pixels are light grey. Pixels with a solution outside the physically reasonable range are dark grey. The use of the TIP table has reduced the required CPU by a factor 100.

5. Conclusion

We presented the inversion package JRC-TIP for the RT within the vegetation canopy. The package minimizes the model data misfit including the deviation from prior information on the unknown model parameters via a quasi-Newton method. It uses the second derivative of the misfit to approximate uncertainty ranges for the estimated model parameters. It further propagates uncertainties from parameters to unobserved radiant fluxes via the model's first derivative. All derivative information is provided by AD of the RT code. The generated derivative code is highly efficient and as accurate as the model code itself.

We highlighted a particular setup of the inverse model which estimates seven parameters from two observed radiant fluxes. For this application we introduced the TIP table, which is based on a fine discretization of the observation space. The TIP table contains for every point on the grid in observation space the associated TIP solution, i.e. the posterior parameter estimates including their uncertainty and the value of the misfit function.

We demonstrated the use of the TIP table to devise and discuss a set of extensions of the package that enhance its robustness. The most efficient of these extensions exploits the smooth

dependence of the model parameters and their minimum on the observations. We demonstrated the performance of the enhanced package for a large-scale application that requires to solve hundreds of millions of inverse problems. Not only does the TIP table provides an enormous speed, it also considerably reduces the fraction of failures.

The concepts presented in this paper can be generalized to other inverse problems that fulfil the following requirements. The use of derivatives requires a smooth model. The additional use of AD requires the model to be coded in a programming language that is accessible to an existing AD tool such as Fortran 77-95 or C(++). The additional application of a database such as the TIP table is useful in data fitting problems whenever a sufficiently fine discretization of the data space is computationally feasible.

Acknowledgments

The study is supported by the European Space Agency under ESRIN/contract no 20595/07/I-EC. The authors would like to thank Simon Blessing for help with graphics.

Appendix. Two-stream forward model

In [7] a new version of a two-stream model suited for the simulation of solar radiant fluxes scattered by, transmitted through and absorbed in, a vegetation canopy is developed and validated. This 1D model ensures the correct balance between the scattered (R), transmitted (T) and absorbed (A) radiant fluxes. It applies the two-stream formulation originally established by [20] to solve the black background problem.

In the case of structurally heterogeneous canopies, effective instead of true state variables have to be adopted, which were introduced in section 2. They consist of the LAI, which is the only spectrally invariant quantity, the leaf single scattering albedo ω_l and the asymmetry factor d_l. The fourth quantity is the albedo of the background, r_{bgd}, which is itself defined as the true (by contrast to effective) value and retrieved as such. With $s = (\mathrm{LAI}, \omega_l, d_l, r_{\mathrm{bgd}})$ we denote the quadruple of state variables that enter the forward model for a given wavelength of the solar radiation. Note that, for easing readability, the $\tilde{}$ (tilde) notation was dropped with respect to [7].

In general, the surface albedo is approximated by a simple weighting of two distinct surface albedo types, each associated with an extreme incident radiation field: the directional hemispherical reflectance (DHR), associated with an incident intensity field which is purely collimated, and the $\mathrm{BHR}_{\mathrm{iso}}$, associated with an incident intensity field that is purely isotropic. Our implementation is based on a weight of 1 for $\mathrm{BHR}_{\mathrm{iso}}$ and 0 for DHR, i.e. we have $R = \mathrm{BHR}_{\mathrm{iso}}$.

While it is beyond the scope of this paper to repeat the entire mathematics of [7] that is involved to derive the scattered fluxes, we will briefly sketch the essential equations to compute R. According to the weighting function above, R is given by [7, equation (33)]

$$R(s) = \overline{R_{\mathrm{veg}}^{\mathrm{Coll}}}(s) + r_{\mathrm{bgd}} \cdot \frac{\overline{T_{\mathrm{veg}}^{\mathrm{Coll}}}^2(s)}{1 - r_{\mathrm{bgd}} \cdot \overline{R_{\mathrm{veg}}^{\mathrm{Coll}}}(s)}. \tag{A.1}$$

Here $\overline{R_{\mathrm{veg}}^{\mathrm{Coll}}}(s)$ represents the contribution due to the radiation travelling downwards that has interacted with the vegetation canopy elements only. This is the so-called black background contribution since $R(s) = \overline{R_{\mathrm{veg}}^{\mathrm{Coll}}}(s)$ is the case of a perfectly absorbing background ($r_{\mathrm{bgd}} = 0$).

We follow the approach of [20, equations (12) and (13)] for the two-stream solutions. For the usual range of canopy optical depth values and under the the assumption of a random

distribution of leaf normal orientation, $\overline{R_{\text{veg}}^{\text{Coll}}}$ and $\overline{T_{\text{veg}}^{\text{Coll}}}$ are computed in accordance with [7, equations (B2) and (B3)]:

$$\begin{aligned}\overline{R_{\text{veg}}^{\text{Coll}}}(s) = {} & \frac{\omega_l}{(1-k^2\,\overline{\mu}^2)\left[(k+\gamma_1)\,\exp\left(k\tfrac{1}{2}\,\text{LAI}\right)+\,(k-\gamma_1)\exp\left(-k\,\tfrac{1}{2}\,\text{LAI}\right)\right]} \\ & \times\Bigg[(1-k\,\overline{\mu})\,(\alpha_2+\,k\,\gamma_3)\exp\left(k\,\frac{1}{2}\,\text{LAI}\right) \\ & -(1+k\,\overline{\mu})\,(\alpha_2-\,k\,\gamma_3)\exp\left(-k\,\frac{1}{2}\,\text{LAI}\right) \\ & -2\,k\,(\gamma_3-\alpha_2\,\overline{\mu})\exp\left(-\frac{1}{2}\,\text{LAI}/\overline{\mu}\right)\Bigg]\end{aligned} \tag{A.2}$$

and

$$\begin{aligned}\overline{T_{\text{veg}}^{\text{Coll}}}(s) = \overline{T_{\text{veg}}^{\text{UnColl}}}(s) - {} & \frac{\omega_l\exp\left(-\tfrac{1}{2}\,\text{LAI}/\overline{\mu}\right)}{(1-k^2\,\overline{\mu}^2)\left[(k+\gamma_1)\,\exp\left(k\tfrac{1}{2}\,\text{LAI}\right)+(k-\gamma_1)\exp\left(-k\tfrac{1}{2}\,\text{LAI}\right)\right]} \\ & \times\Bigg[(1+k\,\overline{\mu})\,(\alpha_1+\,k\,\gamma_4)\,\exp\left(k\,\frac{1}{2}\,\text{LAI}\right) \\ & -(1-k\,\overline{\mu})\,(\alpha_1-k\gamma_4)\exp\left(-\,k\,\frac{1}{2}\,\text{LAI}\right) \\ & -2\,k(\gamma_4+\alpha_1\,\overline{\mu})\exp\left(\frac{1}{2}\,\text{LAI}/\overline{\mu}\right)\Bigg]\end{aligned} \tag{A.3}$$

with

$$\begin{aligned}&\overline{\mu} = 0.5/0.705 \\ &\alpha_1 = \gamma_1\,\gamma_4+\gamma_2\gamma_3 \\ &\alpha_2 = \gamma_1\,\gamma_3+\gamma_2\gamma_4 \\ &k = \left(\gamma_1^2-\gamma_2^2\right)^{1/2}.\end{aligned} \tag{A.4}$$

Using the terminology in [20], $\gamma_3(\gamma_4)$ corresponds to the intercepted fraction of direct radiation scattered in the backward (forward) direction, i.e. creating a source term inside the medium, while $\gamma_1(\gamma_2)$ corresponds to the fraction of the scattered radiation which is redirected in the forward (backward) hemisphere. Recall from section 2 that $\omega_l = r_l + t_l$ and $d_l = r_l/t_l$. With $\delta_l = r_l - t_l$, the γ coefficients are given by [7, table 4]

$$\begin{aligned}&\gamma_1 = 2\left(1-\frac{\omega_l}{2}+\frac{\delta_l}{6}\right) \\ &\gamma_2 = 2\left(\frac{\omega_l}{2}+\frac{\delta_l}{6}\right) \\ &\gamma_3 = \frac{2}{\omega_l}\left(\frac{\omega_l}{4}+\overline{\mu}\frac{\delta_l}{6}\right) \\ &\gamma_4 = 1-\gamma_3.\end{aligned} \tag{A.5}$$

The first term on the right-hand side in equation (A.3), $\overline{T_{\text{veg}}^{\text{UnColl}}}(s)$, is derived by the formula

$$\overline{T_{\text{veg}}^{\text{UnColl}}}(s) = \exp\left(-\frac{1}{2}\,\text{LAI}\right)\left[1-\frac{1}{2}\,\text{LAI}+\left(\frac{1}{2}\,\text{LAI}\right)^2\cdot\exp\left(\frac{1}{2}\,\text{LAI}\right)\cdot\Gamma(0,\text{LAI}/2)\right] \tag{A.6}$$

as shown in [7, equation (16)].

Here

$$\Gamma(0, \mathrm{LAI}/2) = \int_{\mathrm{LAI}/2}^{\infty} t^{-1} \exp(-t)\, \mathrm{d}t \tag{A.7}$$

is the incomplete Gamma function that can be evaluated accurately using a continued fraction development for any value of LAI/2.

References

[1] Solomon S, Qin D, Manning M, Chen Z, Marquis M, Averyt K B, Tignor M and Miller H L 2007 *IPCC, 2007: Climate Change 2007: The Physical Science Basis* Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change

[2] GLOBALVIEW-CO_2 2004 Cooperative Atmospheric Data Integration Project—Carbon Dioxide. CD-ROM, NOAA CMDL, Boulder, Colorado (also available on Internet via anonymous FTP to ftp.cmdl.noaa.gov, path: ccg/co2/GLOBALVIEW)

[3] Takahashi T, Wanninkhof R H, Feely R A, Weiss R F, Chipman D W, Bates N, Olafsson J, Sabine C and Sutherland S C 1999 Net sea-air CO_2 flux over the global oceans: an improved estimate based on the sea-air pCO_2 difference *Extended Abstracts of the 2nd Int. CO_2 in the Oceans Symp. (Tsukuba, Japan, 18–22 January 1999)* pp 9–15

[4] Knyazikhin Y, Martonchik J V, Diner D J, Myneni R B, Verstraete M M, Pinty B and Gobron N 1998 Estimation of vegetation canopy leaf area index and fraction of absorbed photosynthetically active radiation from atmosphere-corrected MISR data *J. Geophys. Res.* **103** 32239–56

[5] Griewank A 1989 On automatic differentiation *Mathematical Programming: Recent Developments and Applications* ed M Iri and K Tanabe (Dordrecht: Kluwer) pp 83–108

[6] Voßbeck M, Giering R and Kaminski T 2008 Development and First Applications of TAC++ *Advances in Automatic Differentiation* (*Lecture Notes in Computational Science and Engineering*) ed C Bischof, H M Bücker, P D Hovland, U Naumann and J Utke (Berlin: Springer) pp 187–97

[7] Pinty B, Lavergne T, Dickinson R E, Widlowski J L, Gobron N and Verstraete M M 2006 Simplifying the interaction of land surfaces with radiation for relating remote sensing products to climate models *J. Geophys. Res.* **111** D02116

[8] Pinty B, Lavergne T, Voßbeck M, Kaminski T, Aussedat O, Giering R, Gobron N, Taberner M, Verstraete M M and Widlowski J-L 2007 Retrieving surface parameters for climate models from MODIS-MISR albedo products *J. Geophys. Res.* **112** D10116

[9] Pinty B, Lavergne T, Kaminski T, Aussedat O, Giering R, Gobron N, Taberner M, Verstraete M M, Voßbeck M and Widlowski J-L 2008 Partitioning the solar radiant fluxes in forest canopies in the presence of snow *J. Geophys. Res.* **113** D04104

[10] Clerici M, Voßbeck M, Pinty B, Kaminski T, Taberner M, Lavergne T and Andredakis I 2010 Consolidating the two-stream inversion package (JRC-TIP) to retrieve land surface parameters from albedo products *IEEE J. Sel. Top. Earth Obs. Remote Sens.* (at press) doi:10.1109/JSTARS.2010.2046626

[11] Tarantola A 1987 *Inverse Problem Theory—Methods for Data Fitting and Model Parameter Estimation* (New York: Elsevier)

[12] Gill P E, Murray W and Wright M H 1981 *Practical Optimization* (New York: Academic)

[13] Giering R and Kaminski T 1998 Recipes for adjoint code construction *ACM Trans. Math. Softw.* **24** 437–74

[14] Talagrand O 1991 The use of adjoint equations in numerical modelling of the atmospheric circulation *Automatic Differentiation of Algorithms: Theory, Implementation, and Application* ed A Griewank and G F Corliss (Philadelphia, PA: SIAM) pp 169–80

[15] Giering R, Kaminski T and Slawig T 2005 Generating efficient derivative code with TAF: adjoint and tangent linear euler flow around an airfoil *Future Gener. Comput. Syst.* **21** 1345–55

[16] Wilkinson J H and Reinsch C 1971 *Handbook for Automatic Computation* (*Linear Algebra* vol 2) (Berlin: Springer)

[17] Goldberg D E 1989 *Genetic Algorithms in Search and Optimization* (Reading, MA: Addison-Wesley)

[18] Metropolis N, Rosenbluth A W, Rosenbluth M N, Teller A H and Teller E 1953 Equation of state calculations by fast computing machines *J. Chem. Phys.* **21** 1087

[19] Kirkpatrick S, Gelatt C D and Vecchi M P 1983 Optimization by simulated annealing *Science* **220** 671–80

[20] Meador W E and Weaver W R 1980 Two-stream approximations to radiative transfer in planetary atmospheres: a unified description of existing methods and new improvements *J. Atmos. Sci.* **37** 630–43