

Oracle-based optimization applied to climate model calibration

Cesar Beltran^{a,*}, N. R. Edwards^b, A. B. Haurie^a, J.-P. Vial^a, D. S. Zachary^c

^a Logilab-HEC, University of Geneva, Geneva, Switzerland

Email: {cesar.beltran, alain.haurie, jean-philippe.vial}@hec.unige.ch

^b Climate and Environmental Physics, University of Bern, Bern, Switzerland

Email: edwards@climate.unibe.ch

^c Physics Department, American University of Sharjah, Sharjah, United Arab Emirates

Email: dzachary@aus.ac.ae

In this paper, we show how oracle-based optimization can be effectively used for the calibration of an intermediate complexity climate model. In a fully developed example, we estimate the 12 principal parameters of the C-GOLDSTEIN climate model by using an oracle-based optimization tool, Proximal-ACCPM. The oracle is a procedure that finds, for each query point, a value for the goodness-of-fit function and an evaluation of its gradient. The difficulty in the model calibration problem stems from the need to undertake costly calculations for each simulation and also from the fact that the error function used to assess the goodness-of-fit is not convex. The method converges to a ‘best fit’ estimate over 10 times faster than a comparable test using the ensemble Kalman filter. The approach is simple to implement and potentially useful in calibrating computationally demanding models based on temporal integration (simulation), for which functional derivative information is not readily available.

Keywords: oracle-based optimization, cutting plane method, model calibration, parameter estimation, C-GOLDSTEIN, Proximal-ACCPM

1. Introduction

The objective of this paper is to show how oracle-based optimization can be effectively used in model calibration for intermediate complexity climate models. Following Oliva [1], *model calibration* is defined as (a) the estimation of the model parameters to obtain the best match between observed and simulated behaviour of the phenomena described (*parameter estimation*) and (b) the assessment of these *estimates* by confidence intervals (*estimate assessment*).

Climate models play a central role in the scientific debate concerning anthropogenic climate change, as indicated, for example, in the Intergovernmental Panel on Climate Change (IPCC) reports [2, 3]. These models are used to simulate the earth system response to the temperature forcing due to anthropogenic emissions of greenhouse gases. They include descriptions of atmosphere, ocean, ice and snow cover and precipitation dynamics in different locations on the planet. Model calibration is a vital issue in climate and earth system modelling since even so-called intermediate complexity models, such as the one used in this work, can have a large number of uncertain parameters. More complex models, generally speaking, have correspondingly more, whereas simpler models typically have fewer parameters, but correspondingly greater uncertainty in their values. For climate prediction, in contrast to short-term weather prediction, parameter values, rather than initial conditions, are considered to be the dominant source of uncertainty. This may be true even for the ocean because

interior processes are poorly understood or quantified, even though initial conditions are forgotten only over millennia.

Oliva [1] distinguishes two approaches for model calibration: *optimal filtering* and *model reference optimization*. In optimal filtering approaches, as for example the ensemble Kalman filter (EnKF) approach proposed by Evensen [4], the parameter estimation and the estimate assessment problems are treated simultaneously. These methods require some probabilistic assumptions concerning the prior distribution of unknown parameters and the model measurement errors (represented by a covariance matrix). In model reference optimization approaches, parameter estimation is done as a first step, followed by a second step concerning estimate assessment. In the parameter estimation step the objective is simply to find parameter values that minimize an *error function* (typically weighted least squares) in order to obtain the best fit between observed and simulated data. No a priori probabilistic information is assumed at this stage. In the estimate assessment step, confidence intervals for the parameter estimates can be determined by performing sensitivity analysis of the minimized error function [1].

We may see the model reference optimization approach, applied to a climate model, as an instance of a design problem where one has to choose the values for static parameters influencing the performance of a complex dynamical system. Indeed, the difficulty lies in the non-explicit link that exists between static design parameter values and evaluation of the dynamic system performance. In this article, we propose to use an oracle-based optimization tool (OBOT), Proximal-ACCPM, i.e., the proximal analytic center cutting plane method [5], to deal

* Corresponding author.

with this difficulty. An OBOT proceeds through a sequence of queries where an oracle replies by sending information about the performance indicator value and its gradient w.r.t. design parameter values. We demonstrate the potential of the method by performing parameter estimation for C-GOLDSTEIN [6], which is a simplified physics, low-resolution climate model, with a 3-D ocean, a 2-D atmosphere and a dynamic and thermodynamic sea-ice component. We describe how the parameter estimation step was performed using an OBOT. The extension to estimate assessment via sensitivity analysis will be the subject of another report.

We compare our results to those reported by Hargreaves et al. [6], who calibrated the C-GOLDSTEIN model using an EnKF method.

The paper is organized as follows: In section 2 we briefly introduce the C-GOLDSTEIN climate model. In section 3 we discuss the potential of Proximal-ACCPM for climate model calibration. In section 4 we formulate the minimization problem used to estimate the C-GOLDSTEIN parameters. In section 5 we briefly introduce Proximal-ACCPM. In section 6 we give the implementation details. In section 7 we report the results and finally, conclusions are given in section 8.

2. C-GOLDSTEIN

2.1. The model

In the hierarchy of climate and earth system models, C-GOLDSTEIN is of intermediate complexity. Owing to a combination of low spatial resolution and simplified physics, the model achieves an integration speed of 1000 or 2000 years per hour on a modern PC (Pentium IV, 2.4 GHz), making it 3 or 4 orders of magnitude (o.o.m.) more computationally efficient than widely used high-resolution general circulation model (GCMs) such as HadCM3 [7], and 1 or 2 o.o.m. faster than other intermediate complexity models with three-dimensional ocean components such as ECBILT-CLIO [8] or the UVic model [9]. On the other hand C-GOLDSTEIN is 1 o.o.m. slower than the reduced dimensionality Bern 2.5-D model [10].

The oceanic momentum budget is represented by a simplified frictional geostrophic balance which is approximately valid for long timescales, of years to decades or more, and large spatial scales, of the order 1000 km or more. The detailed dynamics of oceanic eddies, for instance, are neglected. The atmosphere has a single layer so that atmospheric processes are represented by a balance of energy and moisture plus simple horizontal transport by anisotropic diffusion and advection by a fixed wind field. Feedback involving changes in atmospheric circulation and precipitation patterns, and feedback involving the land surface, are therefore relatively poorly represented or ignored. Sea-ice height and areal coverage are similarly governed by a local heat and fresh water balance, plus

advection by surface currents with a diffusive term to represent unresolved processes. For the studies described here, the ocean component is configured with eight vertical levels while all components share the same 36×36 -cell horizontal grid.

Processes relevant to global-scale ocean circulation are reasonably well represented by this version, as shown in [6]. As a result of its efficient but relatively faithful representation of large-scale ocean dynamics, the model has proved useful for studies of glacial circulation states [11], integrated assessments of climate change impacts [12] and parametric investigations of the stability of the thermohaline circulation [13]. The model is described more fully in [14]. C-GOLDSTEIN forms a component of the Grid Enabled Integrated Earth System Model (GENIE) project (www.genie.ac.uk) in the context of which an Earth System Model with more detailed representations of atmosphere, land ice, ocean biogeochemistry and land-surface processes is under development.

2.2. The parameters

C-GOLDSTEIN contains a total of about 75 physical and model parameters, a subset of 12 which were identified in [14] as the principal adjustable parameters governing transport and mixing, and thus the large-scale distributions of climatic variables (temperature, humidity, etc.). Even where these parameters correspond to well-defined physical processes which may, in principle, be measurable, the correct physical values remain hard to ascertain experimentally. Furthermore, even where an appropriate true global average value could be tightly constrained by measurements, it may be appropriate to allow the model value to deviate from the measured bounds if this allows the model dynamical system to better approximate the real climate, in some desired averaged sense.

Thus in [14], an averaged error function is defined which measures the mean square departure of the model state from observations. Good estimates of the 12 parameters are found by minimizing the error, over an ensemble of 1000 randomly chosen parameter sets within a predefined range. To better compare estimation results, exactly the same optimization problem, using the same data, the same cost function and, as far as possible, the same prior ranges, were addressed both in the present work and in the EnKF study of Hargreaves et al. [6]. However, prior information and cost are treated somewhat differently by Hargreaves et al. [6], as noted below. The observational data correspond to ocean and surface atmospheric statistics averaged over a period of around 50 years. Although there has been significant change in upper ocean and atmosphere temperatures over this period, these changes will be small relative to the error in such a simple model. Indeed, the error function essentially tests how well the model can reproduce the spatial distributions of atmospheric and oceanic variables. Since sources of internal variability such as eddies are not represented, the model typically

responds to the steady imposed solar forcing used here by a slow approach toward a steady state. Physically, then, we consider this steady state as a representation of the preindustrial climate and wish to choose parameters that minimize the deviation between the steady state and the data.

3. Potential of Proximal-ACCPM for climate model calibration

In this section we briefly review possible alternatives for performing model calibration and we indicate why we believe an OBOT such as Proximal-ACCPM is well suited to this task.

As indicated in section 1, model calibration has been attempted via two main strands of approaches: those related to *optimal filtering*, represented by the Kalman filter family of methods, and *model reference optimization*. The Kalman filter can be seen as a recursive *predictor-corrector* method [15] that was initially designed to estimate the state of a stochastic linear dynamical system. By starting at an initial guess of the system state and of the error covariance matrix and after some predictor-corrector iterations, the Kalman filter outputs an optimal estimated state and its associated error covariance matrix. The Kalman filter supposes an underlying linear system that at each iteration *predicts* the future state. Furthermore, at each Kalman filter iteration, the state prediction and its associated error covariance matrix are *corrected* by incorporating the measured information. Both linear system and measurements have associated noise functions which are assumed white and Gaussian.

A powerful variant of the Kalman filter suitable for non-linear systems is the Ensemble Kalman Filter (EnKF), introduced by Evensen [16]. EnKF is based on an ensemble (set) of model states instead of a single model state (in contrast with the Kalman filter). As pointed out in [15], the EnKF can be interpreted as a statistical Monte Carlo method where the ensemble of model states evolves in the state space with the mean as the best estimate and the spreading of the ensemble determining the error variance. In the Kalman filter, the stochasticity of the system state is completely represented by the estimate itself and the error covariance matrix. In contrast, in the EnKF, the stochasticity is approximately represented by the estimate itself and a cloud of points around it (ensemble of system states).

The Kalman filter is also used for parameter estimation (model identification). Anderson [17] shows how to implement the EnKF for this purpose. The technique consists in extending the definition of the state vector so as to include the model parameters (see [18]). Hargreaves et al. [6] report on a calibration of the C-GOLDSTEIN model using the EnKF technique. This study showed a substantial improvement in computational efficiency compared to simpler Monte Carlo parameter estimation techniques, but the calculation process was still relatively

expensive, requiring 54 model runs, each of around 10,000 years (several times the model's intrinsic adjustment timescale). A critical detail of the method was that the model state and model parameters were treated differently in the iteration, in that the prior estimates for the model state were continually updated, whereas the prior estimates for the parameters were not. This constrains the parameters to remain relatively close to the mean of the original, Gaussian parameter distributions.

In model reference optimization approaches, one must first decide on the error function f to be minimized. A common choice is weighted least squares, which can be related to the maximum-likelihood method under appropriate assumptions on the distribution of errors [1]. To minimize the error function, one may use zero-order methods, which only require evaluations of f , first-order methods which also require the evaluation of the gradient g and second-order methods which ask in addition for the evaluation of the Hessian H . In [19], zero-order methods (genetic algorithms, controlled random search, etc.) are used to calibrate different models arising in water industry systems. In this implementation, the number of function evaluations is of the order of several thousands. In [20], a second-order method based on the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method is used to calibrate three parameters of a soil constitutive model. In this application only three parameters had to be estimated and the number of calls to the function evaluation was of the order of 50 to several hundreds.

Proximal-ACCPM is a cutting plane method which belongs to the class of first-order methods. In the C-GOLDSTEIN calibration problem, the only available functional information is the error function value f obtained at a high computational expense. The remaining functional information, g and H , when needed, has to be numerically computed by finite differences or automatic differentiation techniques. In our opinion, a first-order method represents a priori a good compromise when implemented through an algorithm like Proximal-ACCPM, which keeps the number of oracle calls (defining the cutting planes) at a moderate level. Furthermore, most of the optimization methods are based on line search along an improvement direction, which requires extra function evaluations. In our case, the use of a method which is free of line search, like Proximal-ACCPM, is an advantage considering the very high computational cost of each function evaluation.

The main limitation of cutting plane methods is that in principle they are designed for convex optimization. Given that our error function shows non-convexities, we have adapted Proximal-ACCPM in order to deal with them. It is therefore important to note that we will not be able to guarantee that even a local optimum has been reached. However, the method will provide solutions that improve the fit with observed data. We also note that global optimization methods [21], which are designed for non-convex functions, are computationally too demanding for this class of problems. Indeed, these methods rely on enumeration schemes, e.g., branch-and-bound methods,

which usually result in a very high number of function evaluations.

4. The minimization problem

Our goal in this section is to formulate the parameter estimation minimization problem for the C-GOLDSTEIN model. For this purpose, we characterize the modelled climate state by the vector $s \in \mathbb{R}^N$, which consists of the values of oceanic temperature and salinity and of atmospheric temperature and humidity at every model grid point. The minimization therefore does not explicitly consider velocity and sea-ice variables, but these are tightly linked to s by the model dynamics. Note that s corresponds to the steady state attained after a long integration and depends on 12 model parameters represented by x . Unless otherwise stated, for every possible vector of parameters x , C-GOLDSTEIN returns a vector $s(x)$. On the other hand, the real earth climate is represented by the observed state $S \in \mathbb{R}^N$. As error function, we use the weighted mean square error

$$f(x) = \frac{1}{N} \sum_{i=1}^N \frac{(s_i(x) - S_i)^2}{\hat{\sigma}_i^2}, \quad (1)$$

where the weight $\hat{\sigma}_i^2$ is the variance associated to S_i , $i = 1, \dots, N$. There is thus one value of $\hat{\sigma}_i$ for each of the four physical variables, temperature and salinity in the ocean, and temperature and humidity in the atmosphere. Both the variances and the mean error are calculated in computational space rather than physical space, i.e. unweighted by grid-cell volume, so that variables in regions of lower resolution do not carry greater weight. The error function is

thus a balanced measure of how well the model succeeds in representing the quantities it attempts to represent.

Our parameter estimation problem searches for a set of parameters x^* that minimizes f over all allowed parameter sets and can be formulated as

$$\begin{aligned} x^* &= \arg \min f(x) \\ \text{s.t. } x &\in D \subset \mathbb{R}^{12}, \end{aligned} \quad (2)$$

where the box domain D for the model parameters is defined a lower and an upper bound value for each parameter x_i , that is,

$$D = [\underline{x}_1, \bar{x}_1] \times \dots \times [\underline{x}_{12}, \bar{x}_{12}]. \quad (3)$$

The bound values can be found in table 2.

As in any minimization problem, a very important question is whether f is a convex function. However, the complexity of the C-GOLDSTEIN model prohibits us from determining the convexity of $f(x)$ analytically. As a heuristic approach, we can plot slices of the graph of $f(x)$ along the coordinate axes, i.e., we can plot $\phi_i(\alpha) = f(x^0 + \alpha e_i)$, $i = 1, \dots, 12$, where $e_i = (0, \dots, 1_i, \dots, 0)$ is the i th canonical vector of \mathbb{R}^{12} and x^0 is the central point of the domain D . We have encountered convex and non-convex plots (see figures 1 and 2), thus f is clearly not a fully convex function. Similar, single-parameter bifurcation experiments were conducted by Edwards and Marsh [14] for all 12 parameters. Most yielded convex graphs, but parameters controlling the global hydrological cycle in particular, such as atmospheric humidity transport, showed evidence of non-convexity associated with hysteresis and bifurcations between contrasting global ocean circulation states. We shall discuss this matter later. The second main

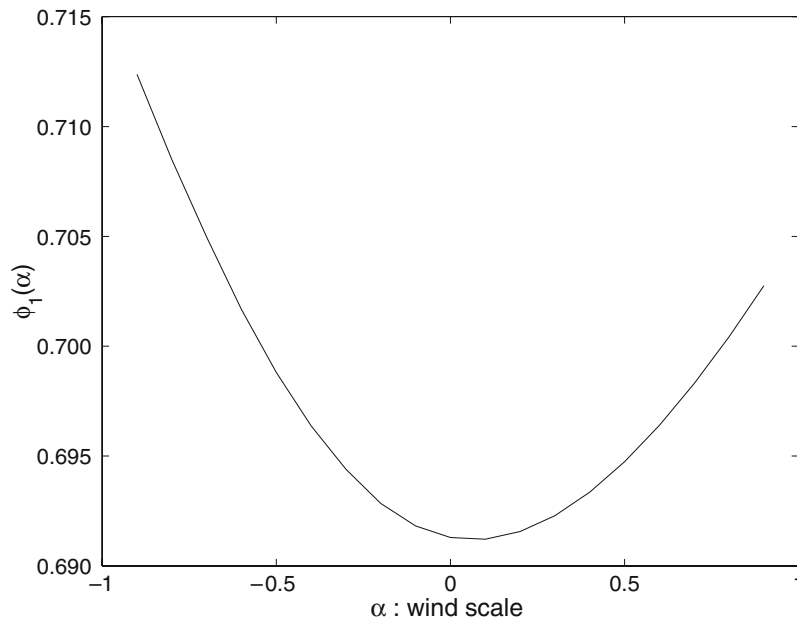


Figure 1. Evolution of $f(x)$ along the wind-scale parameter direction.

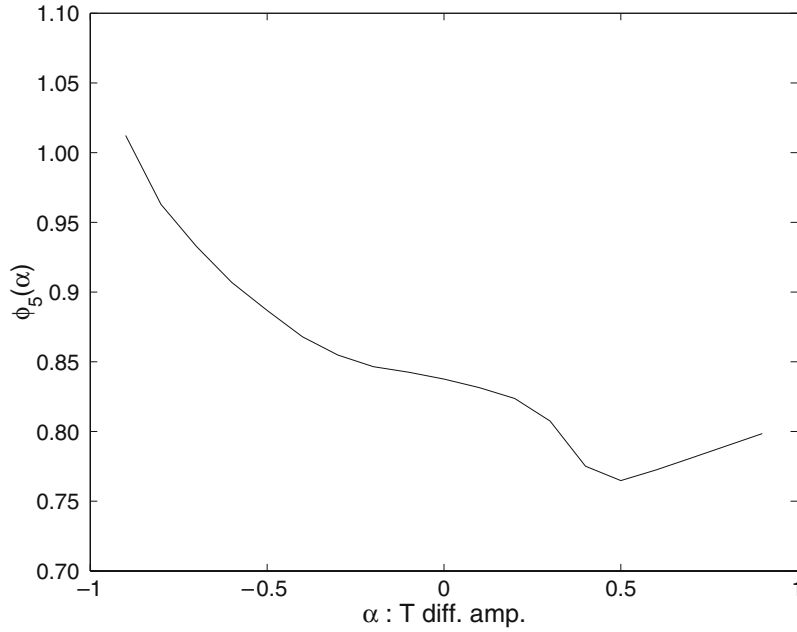


Figure 2. Evolution of $f(x)$ along the T diff. amp. parameter direction.

difficulty is that each evaluation of f takes about 2.5 h on a modern PC (Pentium IV, 2.4 GHz), since for any set of parameters x , $f(x)$ is obtained after a long simulation.

5. Proximal-ACCPM

To solve (2), we use Proximal-ACCPM, an effective and robust cutting plane method [5]. In its iterations, Proximal-ACCPM produces an outer polyhedral approximation of the function epigraph.¹ This approximation serves to delineate a localization set which contains the optimum. For convex optimization problems, the localization set rapidly shrinks to yield an optimal solution x^* . The role of Proximal-ACCPM is to efficiently guide the construction of such an approximation and eventually to find x^* . Proximal-ACCPM is a convex optimization tool and must therefore be applied with care when one deals with a non-convex problem as is the case here.

In the procedure, we consider a sequence of points $\{x^k\}_{k \in K}$ in the search domain D . We denote by g^k the gradient of $f(x)$ at x^k , that is, $g^k = \nabla f(x^k)$. We consider the linear approximation to $f(x)$ at x^k , given by $f^k(x) = f(x^k) + g^k \cdot (x - x^k)$ and have

$$f^k(x) \leq f(x)$$

for all x (to introduce Proximal-ACCPM we assume that f is convex).

The point x^k is referred to as a *query point*, and the procedure to compute the objective function and its

gradient at a query point is called an *oracle*. Furthermore, the hyperplane approximating the objective function $f(x)$ at a feasible query point and defined by the equation $z = f^k(x)$, is referred to as a *cut*.

An upper bound to the minimum value of $f(x)$ is provided by:

$$\theta_u = \min_k f(x^k).$$

The *localization* set is defined as

$$L = \{(x, z) \in \mathbb{R}^{12+1} \mid z \geq f^k(x) \quad \forall k \in K, \quad z \leq \theta_u\}. \quad (4)$$

The basic iteration of a cutting plane method can be summarized as follows

1. Select (\hat{x}, \hat{z}) in the localization set L .
2. Call the oracle at \hat{x} . The oracle returns one cut and a new upper bound $f(\hat{x})$.
3. Update the bounds:
 - (a) $\theta_u \leftarrow \min\{f(\hat{x}), \theta_u\}$.
 - (b) Compute a lower bound θ_l to the optimum of problem (2). For example, $\theta_l = \min\{z \mid (x, z) \in L \cap D\}$.
4. Update the upper bound θ_u and add the new cut in the definition of the localization set (4).

These steps are repeated until a point is found such that $\theta_u - \theta_l$ falls below a prescribed optimality tolerance. The initial domain is thus ‘cut’ down in size by the repeated removal of regions which cannot contain the optimum (given the assumption of convexity).

Cutting plane methods essentially differ in the way one chooses the query point. For instance, the intuitive choice

¹ The set of points that lie on or above the graph of a real-valued function, that is, the set of points $(x, z) \in \mathbb{R}^{n+1}$ such that $z \geq f(x)$.

of the Kelley point (\hat{x}, \hat{z}) [22] that minimizes z in the localization set may prove disastrous, because it over-emphasizes the global approximation property of the localization set. Safer methods, as for example bundle methods [23] or Proximal-ACCPM [5], introduce a regularizing scheme to avoid selecting points too “far away” from the best recorded point. Proximal-ACCPM selects the *proximal analytic center* of the localization set. Formally, the proximal analytic center is the point (\hat{x}, \hat{z}) that minimizes $F_\rho(x, z)$, defined as the logarithmic barrier² of the localization set plus a quadratic proximal term which ensures the existence of a unique minimizer³ for $F_\rho(x, z)$. This point is relatively easy to compute using the standard artillery of Interior Point Methods. Furthermore, Proximal-ACCPM is robust, efficient and particularly useful when the oracle is computationally costly – as is the case in this application.

6. Implementation details

6.1. Variable scaling

The bounds for the 12 parameters \underline{x}_i and \bar{x}_i , which define D in (3), vary greatly from parameter to parameter (see table 2); thus in order to optimize efficiently, x needs scaling. Gill et al. [24] proposed a linear transformation of the form $x = T(y) = Ay + b$, where $A = [a_{ij}]$ is a diagonal matrix, $a_{ii} = 0.5(\bar{x}_i - \underline{x}_i)$, and $b_i = 0.5(\bar{x}_i + \underline{x}_i)$. In this case, $T^{-1}(x)$ transforms domain D into $[-1, 1]^{12}$ and problem (2) is transformed into the following equivalent one:

$$\begin{aligned} y^* &= \arg \min \hat{f}(y) \\ \text{s.t. } y &\in [-1, 1]^{12}, \end{aligned} \quad (5)$$

where

$$\hat{f}(y) = \frac{1}{N} \sum_{i=1}^N \frac{(\hat{s}_i(y) - S_i)^2}{\hat{\sigma}_i^2},$$

and with $\hat{s}(y) = s(T(y)) = s(x)$. Although in practice we solve the scaled problem (5), we will continue to use the simpler ‘ $f(x)$ ’ notation instead of ‘ $\hat{f}(y)$ ’ for convenience.

6.2. Derivative approximation

Since it is not feasible to derive an analytical expression for $\nabla f(x)$, we approximate the partial derivatives by the forward-difference formula:

$$\frac{\partial f(x)}{\partial x_i} \simeq \frac{1}{h} \left[f(x + he_i) - f(x) \right] \quad i = 1, \dots, 12. \quad (6)$$

² The logarithmic barrier for the half space $\{x \in \mathbb{R}^n | a^T x \geq b\}$ is $-\log(b - a^T x)$.

³ That is, the proximal analytic center of L is the point

$$(\hat{x}, \hat{z}) = \arg \min_{x,z} F_\rho(x, z) = \arg \min_{x,z} \left\{ F_L(x, z) + \rho \|x - \tilde{x}\|^2 \right\},$$

where $F_L(x, z)$ is the logarithmic barrier for the localization set L , ρ is the proximal weight, and \tilde{x} is the proximal point (current best point).

As we will see later, we only use approximate values $\tilde{f}(x)$ instead of the true value $f(x)$. In this case, special care must be taken in the computation of the approximated partial derivatives [24]. Given that $\tilde{f}(x)$ introduces an error $\delta(x)$, i.e., $\tilde{f}(x) = f(x) + \delta(x)$, then we have

$$\begin{aligned} \frac{1}{h} \left[\tilde{f}(x + he_i) - \tilde{f}(x) \right] &= \frac{1}{h} \left[f(x + he_i) - f(x) \right] \\ &\quad + \frac{1}{h} \left[\delta(x + he_i) - \delta(x) \right] \\ &\simeq \frac{\partial f(x)}{\partial x_i} + \frac{1}{h} \left[\delta(x + he_i) - \delta(x) \right], \\ &i = 1, \dots, 12. \end{aligned}$$

Thus, even for a small error $\delta(x + he_i) - \delta(x)$, we may obtain completely meaningless approximated derivatives for small values of h . In our computational experience, this has been the case for $h \leq O(10^{-6})$ and best results have been obtained with $h = O(10^{-2})$ or $h = O(10^{-3})$.

6.3. Coping with non-convexity

As already stated, Proximal-ACCPM is designed for convex problems. The convexity assumption is crucial in the convergence analysis of the method [25, 26], and even in its definition. Indeed, the definition of analytic center applies to convex sets with a nonempty interior. The localization set that Proximal-ACCPM builds satisfies this assumption when the function to be minimized is convex. If the function is not convex, simple examples show that the oracle may produce cuts that totally exclude the current localization set. After adding the new cut, the set becomes empty and the method fails.

In our case, we know that the function we minimize is not convex. Indeed, we have observed that the oracle occasionally produces cuts that exclude a previously computed point, which, by construction, belongs to the epigraph set. To cope with the risk of an empty localization set, we use a simple device, based on the observation that the epigraph of a function is unbounded along the vertical axis. To ensure non-emptiness, we check whether a new cut excludes our reference point (the best point in the epigraph generated so far). If not, we proceed as usual, otherwise we lift the upper bounding cut by a sufficient amount.

This procedure is heuristic. It guarantees that the method does not stop unduly. However, we cannot guarantee that the point at which the convergence criterion of Proximal-ACCPM is met corresponds to a local minimum. We can only claim that we have empirical evidence on our climate problem that our simple heuristic device enables Proximal-ACCPM to converge to a point with low least-squares residual. This positive result is probably due to the fact that the function we minimize is only mildly non-convex, at least in the area of interest. Further enhancement of the Proximal-ACCPM technique would be required in order to guarantee convergence to a local minimum in non-convex cases.

6.4. Fast computing of the error function

We now consider how to accelerate the computation of the costly $f(x)$. The error function $f(x)$ is a continuous function of $s(x)$ whose definition is based on the integration of a system of partial differential equations along an infinite time horizon. If we define $\tilde{s}(s^0, t, x)$ as the final state associated to an initial state s^0 , an integration length t and a vector of system parameters x , then by definition $s(x) = \lim_{t \rightarrow \infty} \tilde{s}(s^0, t, x)$ for any $s^0 \in S^0$, the set of suitable initial states of the dynamical system. (Note that we are assuming, for now, that $s(x)$ does not depend on the initial state s^0).

In practice, we have observed that with an integration length of 5000 years, we obtain a very good approximation of $s(x)$ for a reasonable initial state s^0 , i.e., $s(x) \simeq \tilde{s}(s^0, 5000, x)$. Computing of $\tilde{s}(s^0, 5000, x)$, takes about 2.5 h on a modern PC (Pentium IV, 2.4 GHz). At each iteration of Proximal-ACCPM, we need to compute $f(x^k)$ and the approximation to $\nabla f(x^k)$ given by (6), which implies the evaluation of $s(x)$ at 13 points. This means that with this direct approach, we need about 32.5 h of CPU for each Proximal-ACCPM iteration.

An alternative to this direct approach is to use the so-called *warm start* procedure in C-GOLDSTEIN. Assuming, once again, that for two different initial states, s^0 and s^1 , we have

$$s(x) = \lim_{t \rightarrow \infty} \tilde{s}(s^0, t, x) = \lim_{t \rightarrow \infty} \tilde{s}(s^1, t, x),$$

we can accelerate the computing of, say $s(x^b)$, provided we already know $s(x^a)$ for an x^b close to x^a . In this case, $s(x^a)$ should not be too far from $s(x^b)$ and therefore a short integration with $s(x^a)$ as initial state, should be enough to compute $s(x^b)$. That is,

$$s(x^b) \simeq \tilde{s}(s^0, t_s, x^b) \simeq \tilde{s}(s(x^a), t_w, x^b),$$

with $t_w \ll t_s = 5000$. $\tilde{s}(s^0, t_s, x^b)$ and $\tilde{s}(s(x^a), t_w, x^b)$ are respectively called the *standard start* and the *warm start* approximations of $s(x^b)$. Analogously, t_s and t_w are respectively called the standard start and warm start integration lengths.

The error function associated to $\tilde{s}(s^0, t, x)$ is

$$\tilde{f}(s^0, t, x) = \frac{1}{N} \sum_{i=1}^N \frac{(\tilde{s}_i(s^0, t, x) - S_i)^2}{\hat{\sigma}_i^2}.$$

By continuity, $\tilde{f}(s^0, t, x)$ inherits the asymptotic behaviour of $\tilde{s}(s^0, t, x)$. For example, it is easy to see that $f(x) = \lim_{t \rightarrow \infty} \tilde{f}(s^0, t, x)$ for any $s^0 \in S^0$. Analogously, $\tilde{f}(s^0, t_s, x^b)$ and $\tilde{f}(s(x^a), t_w, x^b)$ are respectively called the standard start and the warm start approximations of $f(x^b)$.

To compute fast and reliable approximation to $f(x^k)$ and to $\nabla f(x^k)$, we use the warm start procedure as follows:

Central point warm start algorithm

1. *Warm start initialization*: prior to any Proximal-ACCPM iteration, select $s^0 \in S^0$. Compute and store

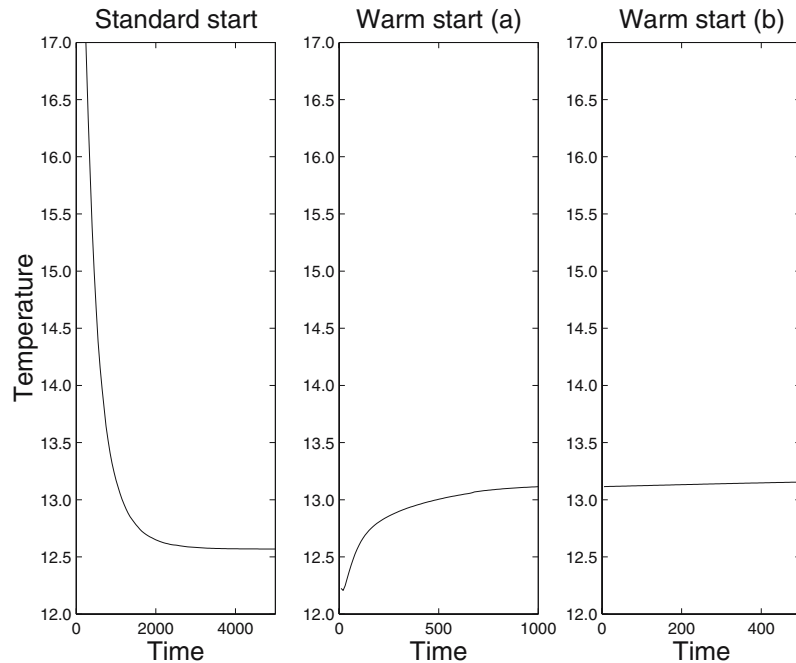


Figure 3. Deepwater temperature in the model Atlantic as a function of time for various initial states. Left-hand side plot: The standard start needs a long integration to attain a steady state. Central plot: The warm start needs a shorter integration to attain a steady state. Right-hand side plot: The warm start at a slightly perturbed point attains steadiness with a very short integration.

$\tilde{s}^0 = \tilde{s}(s^0, t_s, x^0)$ the standard start approximation of $s(x^0)$.

2. *Warm start*: at each Proximal-ACCPM iteration k compute and store

$$\tilde{s}^k = \tilde{s}(\tilde{s}^{k-1}, t_w, x^k),$$

approximate $f(x^k)$ by $\tilde{f}(\tilde{s}^{k-1}, t_w, x^k)$ and approximate $f(x^k + he_i)$ by $\tilde{f}(\tilde{s}^k, t'_w, x^k + he_i)$, $i = 1, \dots, 12$.

3. *Partial derivatives*: approximate partial derivatives $\partial f(x^k)/\partial x_i$ by:

$$\frac{1}{h} \left[\tilde{f}(\tilde{s}^k, t'_w, x^k + he_i) - \tilde{f}(\tilde{s}^{k-1}, t_w, x^k) \right] \quad i = 1, \dots, 12. \quad (7)$$

4. *Accurate value of $f(x^*)$* : once Proximal-ACCPM determines x^* as the best encountered point, approximate $f(x^*)$ by an extra standard start C-GOLDSTEIN call $\tilde{f}(s^0, t_s, x^*)$.

Note that at step 2, at each perturbed point $x^k + he_i$, the warm start procedure uses the steady state attained when computing the objective function at the central point x^k . This approach takes advantage of the fact that the distance between any of these perturbed points and the central point is h . For this reason, at a perturbed point, by using the warm start procedure, the dynamical system recovers its steadiness after a very short integration length. This phenomenon can be intuitively observed in figure 3.

An alternative warm start approach could be as follows:

Parallel warm start algorithm

1. *Warm start initialization*: prior to any Proximal-ACCPM iteration, select $s^0 \in S^0$. Compute and store $\tilde{s}^{0,i} = \tilde{s}(s^0, t_s, x^0 + he_i)$, the standard start value of $s(x^0 + he_i)$ for $i = 0, \dots, 12$ (the vector e_0 being the null vector).
2. *Warm start*: at each Proximal-ACCPM iteration k , compute and store

$$\tilde{s}^{k,i} = \tilde{s}(\tilde{s}^{k-1,i}, t_w, x^k + he_i)$$

and approximate $f(x^k + he_i)$ by $\tilde{f}(\tilde{s}^{k-1,i}, t_w, x^k + he_i)$, $i = 0, \dots, 12$.

Steps 3 and 4 would be as in the central point warm start algorithm. Obviously, this approach is better suited for parallel computing than the previous one, since now the 13 C-GOLDSTEIN calls at step 2 are independent. The main drawback of the Parallel procedure is that it does not take advantage of the proximity between the central and perturbed points when computing the approximation to the partial derivatives (see next section). Since we are using a single processor PC, this is the reason why we have chosen the central point warm start in this paper.

6.5. Selection of the integration length

The critical parameter in our method is the integration length in the C-GOLDSTEIN model. A long integration will result in accurate results but long computations. A too-short integration will produce the opposite effect. Our aim is therefore to find an integration length which balances both computation time and accuracy. The selection of the integration lengths t_s , t_w and t'_w has been done in a heuristic way by observing the evolution of the climate state. The evolution of the climate state as a function of time $\tilde{s}(s, t, x)$ can be visualized by plotting the basin-averaged deepwater temperature in the model Atlantic (below ≈ 1000 m depth), $T_d(s, t, x)$.

In figure 3, the left-hand plot corresponds to $T_0(t) = T_d(s^0, t, x^0)$ obtained by a standard start C-GOLDSTEIN call with initial state s^0 and x^0 as parameter vector. The initial state s^0 is a globally uniform state with warm water throughout the ocean, as used by Edwards and Marsh [14]. We observe that the temperature stabilizes around 12.6°C after 5000 integration years, i.e., $\lim_{t \rightarrow \infty} T_d(s^0, t, x^0) \simeq T_d(s^0, 5000, x^0) = 12.6^\circ\text{C}$.

The central plot corresponds to $T_1(t) = T_d(\tilde{s}^0, t, x^1)$, obtained by a warm start C-GOLDSTEIN call with initial state $\tilde{s}^0 = \tilde{s}(s^1, 5000, x^0)$ and x^1 as parameter vector. We observe that, by using the warm start, the temperature stabilizes around 13.15°C after an integration period of only 1000 years, i.e., $\lim_{t \rightarrow \infty} T_d(\tilde{s}^0, t, x^1) \simeq T_d(\tilde{s}^0, 1000, x^1) = 13.15^\circ\text{C}$.

The right-hand plot corresponds to $T_2(t) = T_d(\tilde{s}^1, t, x^1 + he_1)$ obtained by a warm start C-GOLDSTEIN call with initial state $\tilde{s}^1 = \tilde{s}(\tilde{s}^0, 1000, x^1)$ and $x^1 + he_1$ as parameter vector. We observe that, by using the warm start at a slightly perturbed point, the temperature stabilizes very quickly. In order to reduce the CPU time, the goal is to take integration lengths t_s , t_w and t'_w as short as possible, but ensuring the steadiness of the C-GOLDSTEIN dynamical system in order to obtain well-approximated values of f and its partial derivatives. From our numerical experience, we have observed that good values for the integration times are: $t_s = 5000$, $t_w = 500$ and $t'_w = 50$. Furthermore, C-GOLDSTEIN routinely calculates a diagnostic parameter ROC which measures the root mean square rate of change of ocean variables and thus provides a very strong measure of unsteadiness. This can be used to guarantee the quality of our approximation to $f(x^k)$ in step 2 of the central point algorithm. After computing $\tilde{f}(\tilde{s}, t, x^k)$, a low value of the associated ROC ensures a good approximation to $f(x^k)$. In our implementation, whenever the ROC has been greater than 0:01, an extra integration has been performed in order to double the integration time and thus improve the approximation to $f(x^k)$. This last mechanism is seldom used and, roughly speaking, we can say that by using this warm start setting, at each Proximal-ACCPM iteration we divide by nearly 60 ($13 \times 5000 / (500 + 12 \times 50)$) the standard start integration time per iteration, without noticeably increasing the number of Proximal-ACCPM iterations required.

7. Results

In this section we compare the results of our model calibration with those of the ensemble Kalman filter (EnKF) method, both from the numerical perspective and from the point of view of the resulting model climate.

7.1. Numerical performance

In our approach, we used $h = 10^{-2}$ to estimate the derivatives [see equation (6)] and $\epsilon_{\text{ACCPM}} = 10^{-3}$ for the Proximal-ACCPM stopping tolerance. As initial point x^0 for Proximal-ACCPM, we took the center of the box domain D in equation (2). The initial state s^0 is globally uniform, as noted above. Programs were written in MATLAB 6.1 [27] and run on a PC (Pentium IV, 2.4 GHz, with 6 Gb of RAM) under the Linux operating system. C-GOLDSTEIN is coded in Fortran 77.

Figure 4 shows the evolution of the Proximal-ACCPM upper and lower bounds up to convergence after 33 iterations. In the case of a convex error function, the upper bound plot would be monotonically non-increasing, that is, $\theta_u^k \geq \theta_u^{k+1}$ for all k . It is not the case here, however, due to the apparently mild non-convexity of f . As shown in Section 6, Proximal-ACCPM may lift the current upper bound in order to cope with the non-convex case.

In terms of solution quality, Proximal-ACCPM finds a slightly lower function value of 0.4895 compared to 0.4944 for the EnKF approach (see tables 1 and 2). Regarding performance, the more reliable integration length (as opposed to the CPU hours) was used to compare the two methods. Proximal-ACCPM converges after 33 iterations, when the relative gap between the upper bound (objective

function) and the lower bound falls below the stopping threshold ϵ_{ACCPM} (see figure 4). The total number of integration years for Proximal-ACCPM after 33 iterations was: 5000 years for the warm start initialization, plus $33 \times (500 + 12 \times 50)$ for the Proximal-ACCPM iterations (warm start), plus 700 integration years to improve the ROC parameter in a few cases plus 5000 integration years to compute the exact value of the error function at the optimal point. All in all, we needed 47,000 integration years compared to around $54 \times 10,000 = 540,000$ integration years for the EnKF method, as reported in [6]. The overhead CPU times for Proximal-ACCPM and the EnKF methods are both considered negligible compared to the climate integration times. For the computer we used, the reported integration lengths would correspond to estimated CPU times of 270.0 and 23.5 hours for EnKF and Proximal-ACCPM methods, respectively. It should be noted that the large amount of data processed in the EnKF method can lead to further computational overheads; however, EnKF computation can be parallelised, corresponding to a minimum integration time of 5 CPU hours per node across 54 nodes. In contrast to the results obtained by EnKF and Proximal-ACCPM, the Latin hypercube Monte Carlo method of Edwards and Marsh [14], required 2,000,000 integration years, but failed to locate any solutions with error less than 0.6000.

It is worth mentioning that it is common practice to test a parameter calibration method by an identical twin test in which an arbitrary system state is used as artificial data and the calibration method attempts to determine what parameter values were used to produce it. Indeed, the EnKF development for C-GOLDSTEIN followed this approach, with Annan et al. [28] showing that EnKF was able to

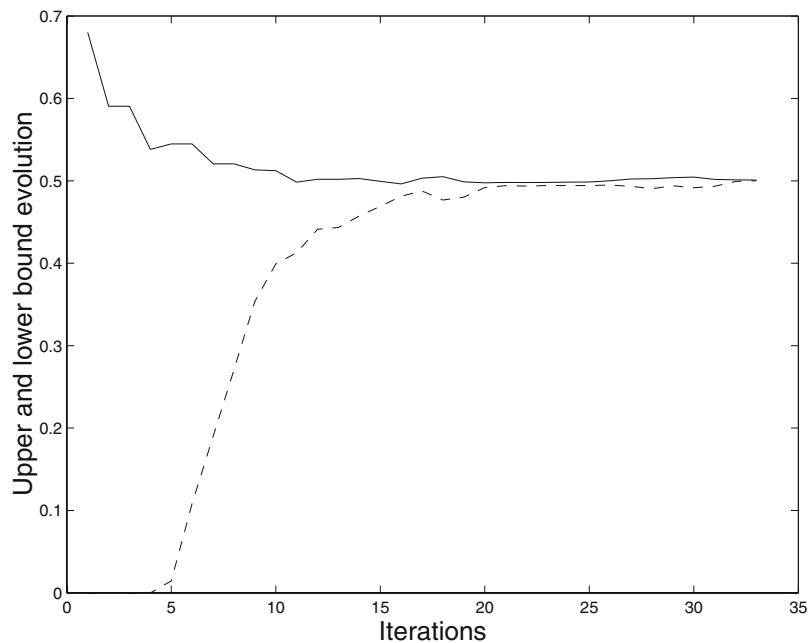


Figure 4. Proximal-ACCPM convergence.

Table 1
Performance.

Method	Lowest error	Integration length (years)
EnKF	0.4944	540,000
Proximal-ACCPM	0.4896	47,000

Note that the EnKF value corresponds to 10,000 years per ensemble member and that the estimate referred to corresponds to the ensemble mean, which may not be the lowest-error solution found.

solve the identical twin problem in around $n \times 2000$ years, where the ensemble size n in the test was set to 54 (their figure 2), and Hargreaves et al. [6] showing that with real data the integration cost increased to around $n \times 10,000$ years (from their figure 4). In the case of Proximal-ACCPM, we found that if the artificial data were used as an initial state, the method was able to find the exact parameters used to extremely high accuracy in only around 100 Proximal-ACCPM iterations. With a very short integration period of 2 years, this gave a total integration time for estimation of the 12 parameters of less than 3000 years. In other words, Proximal-ACCPM was able to solve the identical twin test so efficiently that it was necessary to proceed directly to the case with real data to have a meaningful assessment of the method.

7.2. Model climate

As expected from the decrease in mean error, the final state $\tilde{s}(x^*)$ of the optimization procedure constitutes a significantly more realistic climate than the initial guess $\tilde{s}(x^0)$. We do not discuss the model climate in detail here because the final state is similar to that obtained using the EnKF procedure and discussed in detail by Hargreaves et al. [6]. By way of illustration we show, in figure 5, the sea-surface temperature (SST) field in the final state, along with the

SST difference fields between this state and the initial guess, the EnKF solution, and the data. Changes of several degrees are visible compared to the initial state, leading to broad regions of low error compared to the data. Large systematic errors remain, however, particularly in the regions of the cold upwelling pools in the eastern tropical Pacific and Atlantic and in the boundary current separation regions of the Gulf Stream and Kuroshio. These regions are challenging even for much more computationally expensive climate models. The EnKF solution has very similar SST and thus similar systematic errors, which it is reasonable to assume are inherent to the model dynamics. It is noteworthy that the method is able to find a minimum-error state without difficulty, even when the optimal model climate is relatively far removed from the observational target.

Most intriguing is that the EnKF and Proximal-ACCPM solutions are not exactly the same. Indeed, there are substantial differences in most parameters, as indicated in table 2. According to the error estimate provided by the EnKF method, the parameter values obtained using Proximal-ACCPM are, on average, around 2 standard deviations away from the mean obtained using the EnKF (see [6]). That the solutions are different is to be expected since, as noted above, the EnKF estimate is constrained to remain relatively close to the initial estimate. Furthermore, it was noted by Hargreaves et al. [6] that certain parameters are only poorly constrained, in particular the parameter governing the behaviour of sea ice, for which no data constraint was applied. Sea-ice diffusivity in the Proximal-ACCPM solution is very close to the imposed bound, and experience with both methods suggests that larger values are preferred. The EnKF, however, penalizes solutions which are near to the bounds. Another relevant point is that Hargreaves et al. [6] found certain parameters to be correlated, indicating possible redundancy in the tuned parameter set, in particular the multiple parameters governing atmospheric transport. Closer examination reveals that

Table 2
Estimated parameters.

Parameters	Lower bound	Upper bound	Estimated parameters		Units
			EnKF	Proximal-ACCPM	
<i>Ocean</i>					
Wind-scale	1.0	3.0	1.6674	1.1841	–
Isopyc. diff.	3.0×10^2	1.0×10^4	4.1264×10^3	5.5321×10^3	$\text{m}^2 \text{s}^{-1}$
Diapyc. diff.	2.0×10^{-6}	2.0×10^{-4}	1.8134×10^{-5}	3.8818×10^{-5}	$\text{m}^2 \text{s}^{-1}$
Friction ⁻¹	5.0×10^{-1}	5.0	3.4331	4.9959	days
<i>Atmosphere</i>					
T diff. amp.	1.0×10^6	1.0×10^7	3.7548×10^6	2.5839×10^6	$\text{m}^2 \text{s}^{-1}$
q diff.	5.0×10^4	5.0×10^6	1.7447×10^6	1.9337×10^6	$\text{m}^2 \text{s}^{-1}$
T adv. coeff.	0.0	1.0	6.0357×10^{-2}	8.9163×10^{-2}	–
q adv. coeff.	0.0	1.0	1.3674×10^{-1}	1.4885×10^{-2}	–
Sea-ice diff.	5.0×10^2	8.0×10^3	6.2494×10^3	7.9913×10^3	$\text{m}^2 \text{s}^{-1}$
FWF adjust.	0.0	2.0	8.9796×10^{-1}	1.0406	–
T diff. width	5.0×10^{-1}	2.0	1.3071	1.992	radians
T diff. slope	0.0	2.5×10^{-1}	6.8597×10^{-2}	2.3644×10^{-1}	–

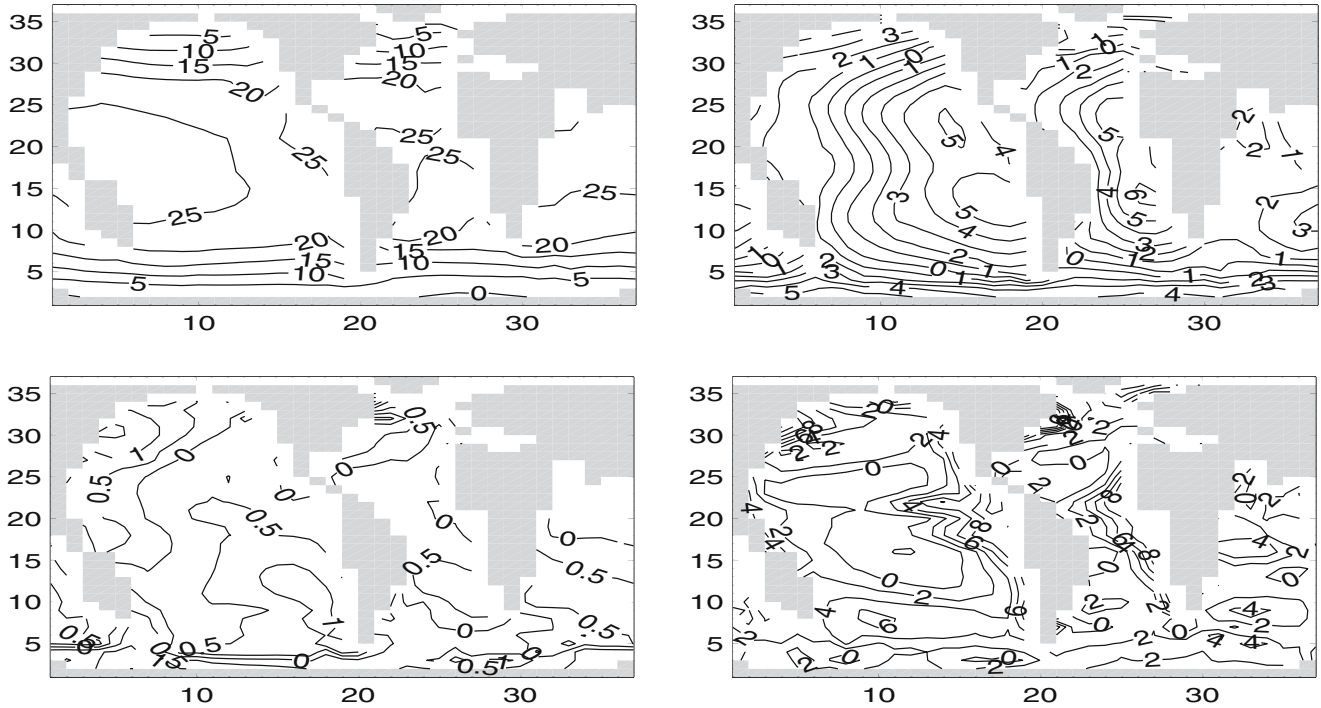


Figure 5. Final sea-surface temperature (SST) field and SST difference fields in Celcius. Upper left, the final state; upper right, final state minus initial state s^0 ; lower left, final state minus EnKF solution; lower right, final state minus data. Plots are in model grid coordinates.

differences in atmospheric transport appear to physically explain the differences in the modelled climate states.

We are thus left with two significantly different sets of “optimal” parameters arising from different approaches to calibration but associated with very similar climate states. Remaining close to the original priors gives a better-posed mathematical problem, but indicates that the result depends on initial guesses for both the mean and the variance of the parameter distributions, rather than simply on the imposed extreme values. It must also be borne in mind, as shown in [13], that in certain regions of parameter space C-GOLDSTEIN exhibits multiple solutions for fixed parameter values, corresponding to qualitatively different steady ocean circulation states. By more devious choice of initialization procedures, the model has been found to possess up to 12 steady solutions for fixed parameters in some regimes, mostly corresponding to qualitatively very similar states (Hargreaves, personal communication). Thus it is likely that a given method could produce different solutions depending on the initial conditions and solution path. Another possibility, which cannot be ruled out, is that the optimization procedure itself could have multiple solutions, in other words, that multiple local minima of the cost function might exist. This would be in line with the results of Edwards and Marsh [14], where the Monte Carlo optimization, effectively a global optimization, while admittedly undersampled, found both good and bad solutions almost throughout the range of each parameter.

Whatever the root cause, the possible existence of different optimal solutions using different methods or

initial conditions indicates that, although great improvements can be obtained in the fit of model to data and thus in the objectivity of the model calibration process, it may be difficult to identify with confidence the globally optimal values for individual parameters. Climate model calibration should thus be viewed as part of a continual process of improving knowledge of model errors as a function of prior assumptions concerning model parameters.

8. Conclusions

In this paper, we showed how oracle-based optimization can be effectively used in the calibration of intermediate complexity climate models. In a fully developed example, the 12 principal transport and mixing parameters of the C-GOLDSTEIN climate model were estimated by using Proximal-ACCPM as the oracle-based optimization tool. In terms of goodness-of-fit, we obtained estimates of similar quality to those obtained by an EnKF approach, but in around one tenth of the total model integration time. Nevertheless, the two approaches cannot really be compared since the EnKF approach also performs the estimate assessment. The fact that the solution found by Proximal-ACCPM was well outside the estimated error bounds provided by the EnKF should serve as a warning that the results of model calibration can depend on the initial assumptions concerning model parameters. Results may also be subject to sensitivity to initial conditions, or redundancy in the process of minimizing a simple scalar

error function by tuning many related parameters in a model with large systematic errors. Overall, the result is that it may be difficult to identify with confidence the globally optimal values for individual parameters. Nevertheless, optimization is able to produce considerable improvements in model performance and remove at least some of the subjectivity which plagues the model development process. The oracle-based optimization approach used here is highly efficient, simple to implement and generally applicable and could thus be useful in calibrating other computationally demanding models based on temporal integration (simulation), for which functional derivative information is not readily available. Whether the approach could be successfully applied to more chaotic models, such as eddy-resolving climate models, remains untested and uncertain. In [29], it is argued that the pathological behaviour of even averaged statistics in such cases handicaps any derivative-based method, although the derivative-based 4DVar method is widely used for assimilation in eddying models over very short integration periods.

Our contribution has been empirical. By using a well-established optimization tool, Proximal-ACCPM, we have performed a fast tuning of C-GOLDSTEIN, a model of intermediate complexity. The keys for fast tuning have been: (1) the fast computation of the error function and approximated derivatives by exploiting the restart procedure (warm start) in the C-GOLDSTEIN model; (2) the rapid convergence and robust behaviour, particularly in respect of non-convexity, of Proximal-ACCPM.

From the climate modelling point of view, two refinements to the work described here would be highly desirable: firstly, the use of an error function which incorporates information from a variety of oceanic tracers representing different timescales, and, secondly, developing the ability to tune two separate steady states of the model to different data sets. This would make it possible to demand that a model successfully should reproduce more than one climate state. An obvious example would be simultaneous tuning to glacial and interglacial states.

Acknowledgements

The research has been supported by the Swiss NCCR-Climate program.

References

- [1] R. Oliva, Model calibration as a testing strategy for system dynamics models, *Eur. J. Oper. Res.* 151 (2002) 552–568.
- [2] IPCC, Special report on emissions scenarios, working group III. Technical report, Intergovernmental Panel on Climate Change (IPCC) (Cambridge University Press, 2000).
- [3] IPCC, Climate change 2001: the scientific basis, summary for policy-makers and technical summary of the working group I report. Technical report, Intergovernmental Panel on Climate Change (IPCC) (Cambridge University Press, 2001).
- [4] G. Evensen, The ensemble Kalman filter: theoretical formulation and practical implementation, *Ocean Dyn.* 53 (2003) 343–367.
- [5] O. du Merle and J.-P. Vial, Proximal ACCPM, a cutting plane method for column generation and Lagrangian relaxation: application to the p-median problem. Technical report, Logilab, HEC (University of Geneva, 2002).
- [6] J.C. Hargreaves, J.D. Annan, N.R. Edwards and R. Marsh, An efficient climate forecasting method using an intermediate complexity earth system model and the ensemble Kalman filter, *Clim. Dyn.* 23 (2004) 745–760.
- [7] C. Gordon, C. Cooper, C.A. Senior, H. Banks, J.M. Gregory, T.C. Johns, J.F.B. Mitchell and R.A. Wood, The simulation of SST, sea-ice extents and ocean heat transports in a version of the Hadley Centre coupled model without flux adjustments, *Clim. Dyn.* 16 (2000) 147–168.
- [8] H. Goosse, F.M. Selten, R.J. Haarsma and J.D. Opsteegh, Decadal variability in high northern latitudes as simulated by an intermediate-complexity climate model, *Ann. Glaciol.* 33 (2001) 525–532.
- [9] A.J. Weaver, M. Eby, E.C. Wiebe, C.M. Bitz, P.B. Duffy, T.L. Ewen, A.F. Fanning, M.M. Holland, A. MacFadyen, H.D. Matthews, K.J. Meissner, O. Saenko, A. Schmittner, H. Wang and M. Yoshimori, The UVic earth system climate model: model description, climatology, and applications to past, present and future climates, *Atmos.–Ocean* 39 (2001) 361–428.
- [10] T.F. Stocker, D.G. Wright and L. A. Mysak, A zonally averaged, coupled ocean–atmosphere model for paleoclimate studies, *J. Clim.* 5 (1992) 773–797.
- [11] E.J. Rohling, R. Marsh, N.C. Wells, M. Siddall and N.R. Edwards, Similar meltwater contributions to glacial sea level changes from Antarctic and northern ice sheets, *Nature* 430 (2004) 1016–1021.
- [12] L. Drouet, C. Beltran, N.R. Edwards, A.B. Haurie, J.-P. Vial and D.S. Zachary, An oracle method to couple climate and economic dynamics, in: *Coupling Climate and Economic Dynamics*, eds. A. Haurie and L. Viguier (Kluwer, 2005).
- [13] R. Marsh, A. Yool, T.M. Lenton, M.Y. Gulamali, N.R. Edwards, J.G. Shepherd, M. Krznanic, S. Newhouse and S.J. Cox, Bistability of the thermohaline circulation identified through comprehensive 2-parameter sweeps of an efficient climate model, *Clim. Dyn.* 23 (2005) 761–777.
- [14] N.R. Edwards and R. Marsh, Uncertainties due to transport-parameter sensitivity in an efficient 3-d ocean-climate model, *Clim. Dyn.* 24(4) (2005) 415–433.
- [15] G. Welch and G. Bishop, An introduction to the Kalman filter. Technical report, Department of Computer Science, University of North Carolina at Chapel Hill, 2003.
- [16] G. Evensen, Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, *J. Geophys. Res.* 99(C5) (1994) 10143–10162.
- [17] J.L. Anderson, An ensemble adjustment Kalman filter for data assimilation, *Mon. Weather Rev.* 129 (2001) 2884–2902.
- [18] J. Derber, A variational continuous assimilation scheme, *Mon. Weather Rev.* 117 (1989) 2437–2446.
- [19] D.P. Solomatine, *Water industry systems: modelling and optimization applications*, volume 2, chapter Random search methods in model calibration and pipe network (Research studies press, Ltd., Baldock, UK, 1999) pp. 317–332.
- [20] Z. Yang and A. Elgamal, Application of unconstrained optimization and sensitivity analysis to calibration of a soil constitutive model, *Int. J. Numer. Anal. Methods Geomech.* 27 (2003) 1277–1297.
- [21] R. Horst and H. Tuy, *Global Optimization: Deterministic Approaches*, 3rd edn. (Springer Verlag, Heidelberg, 1996).

- [22] J.E. Kelley, The cutting-plane method for solving convex programs, *J. SIAM* 8 (1960) 703–712.
- [23] J.B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Vol. I and II (Springer-Verlag, Berlin, 1996).
- [24] P.E. Gill, W. Murray and M.H. Wright, *Practical Optimization*, 10th edn. (Academic Press, London, 1995).
- [25] Y. Nesterov, Cutting plane algorithms from analytic centers: efficiency estimates, *Math. Program.*, B 69 (1995) 149–176.
- [26] J.L. Goffin, Z.Q. Luo and Y. Ye, Complexity analysis of an interior cutting plane for convex feasibility problems, *SIAM J. Optim.* 6 (1996) 638–652.
- [27] D.J. Higham and N.J. Higham, *MATLAB Guide*, (SIAM, Philadelphia, PA, USA, 2000).
- [28] J.D. Annan, J.C. Hargreaves, N.R. Edwards and R. Marsh, Parameter estimation in an intermediate complexity earth system model using an ensemble Kalman filter, *Ocean Model.* 8 (2005) 135–154.
- [29] J.D. Annan and J.C. Hargreaves, Efficient parameter estimation for a highly chaotic system, *Tellus A*, 2005.