# Simultaneous perturbation stochastic approximation for tidal models

**Muhammad Umer Altaf · Arnold W. Heemink ·
Martin Verlaan · Ibrahim Hoteit**

**Abstract** The Dutch continental shelf model (DCSM) is a shallow sea model of entire continental shelf which is used operationally in the Netherlands to forecast the storm surges in the North Sea. The forecasts are necessary to support the decision of the timely closure of the moveable storm surge barriers to protect the land. In this study, an automated model calibration method, simultaneous perturbation stochastic approximation (SPSA) is implemented for tidal calibration of the DCSM. The method uses objective function evaluations to obtain the gradient approximations. The gradient approximation for the central difference method uses only two objective function evaluation independent of the number of parameters being optimized. The calibration parameter in this study is the model bathymetry. A number of calibration experiments is performed. The effectiveness of the algorithm is evaluated in terms of the accuracy of the final results as well as the computational costs required to produce these results. In doing so, comparison is made with a traditional steepest descent method and also with a newly developed proper orthogonal decomposition-based calibration method. The main findings are: (1) The SPSA method gives comparable results to steepest descent method with little computational cost. (2) The SPSA method with little computational cost can be used to estimate large number of parameters.

## 1 Introduction

Accurate sea water level forecasting is crucial in the Netherlands. This is mainly because large areas of the land lie below sea level. Forecasts are made to support the storm surge flood warning system. Timely water level forecasts are necessary to support the decision for closure of the movable storm surge barriers in the Eastern Scheldt and the New Waterway. Moreover, forecasting is also important for harbor management, as the size of some ships have become so large that they can only enter the harbor during high water period. The storm surge warning service (SVSD) in close cooperation with the Royal Netherlands meteorological institute is responsible for these forecasts. The surge is predicted by using a numerical hydrodynamic model, the Dutch continental shelf model (DCSM) (see Stelling 1984; Verboom et al. 1992). The performance of the DCSM regarding the storm surges is influenced by its performance in forecasting the astronomical tides. Using inverse modeling techniques, these tidal data can be used to improve the model results.

Most efficient optimization algorithms require a gradient of the objective function. This usually requires the implementation of the adjoint code for the com-

M. U. Altaf (✉) · A. W. Heemink · M. Verlaan
Delft University of Technology, Mekelweg 4,
2628 CD, Delft, Netherlands
e-mail: m.u.altaf@ewi.tudelft.nl, umaraltaf74@hotmail.com

I. Hoteit
King Abdullah University of Science and Technology,
25955-690, Thuwal, Saudi Arabia

putation of the gradient of the objective function. The adjoint method aims at adjusting a number of unknown control parameters on the basis of given data. The control parameters might be model initial conditions or model parameters (Thacker and Long 1988). A sizeable amount of research on adjoint parameter estimation was carried out in the last 30 years in fields such as meteorology, petroleum reservoirs, and oceanography for instance by Seinfeld and Kravaris (1982), Bennet and Mcintosh (1982), Ulman and Wilson (1998), Courtier and Talagrand (1990), Lardner et al. (1993) and Heemink et al. (2002). A detailed description of the application of the adjoint method in atmosphere and ocean problems can be found in Navon (1998).

One of the drawbacks of the adjoint method is the programming effort required for the implementation of the adjoint model. Research has recently been carried out on automatic generation of computer code for the adjoint, and adjoint compilers have now become available (see Kaminski et al. 2003). Even with the use of these adjoint compilers, this is a huge programming effort that hampers new applications of the method. Courtier et al. (1994) had proposed an incremental approach, in which the forward solution of the nonlinear model is replaced by a low resolution approximate model. Reduced order modeling can also be used to obtain an efficient low-order approximate linear model (Hoteit 2008; Lawless et al. 2008).

This paper focuses on a method referred to as the simultaneous perturbation stochastic approximation (SPSA) method. This method can be easily combined with any numerical model to do automatic calibration. For the calibration of numerical tidal model, the SPSA algorithm would require only the water level data predicted from the given model. SPSA is stochastic offspring of the Keifer–Wolfowitz Algorithm (Kiefer and Wolfowitz 1952) commonly referred as finite difference stochastic approximation (FDSA) method. This algorithm uses objective function evaluations to obtain the gradient approximations. Each individual model parameter is perturbed one at a time and the partial derivatives of the objective function with respect to the each parameter is estimated by a divided difference based on the standard Taylor series approximation of a partial derivative. This approximation of each partial derivative involved in the gradient of the objective function requires at least one new evaluation of the objective function, thus this method is not feasible for automated calibration when we have large number of parameters.

The SPSA method uses stochastic simultaneous perturbation of all model parameters to generate a search at each iteration. SPSA is based on a highly efficient and easily implemented simultaneous perturbation approximation to the gradient. This gradient approximation for the central difference method uses only two objective function evaluation independent of the number of parameters being optimized. The SPSA algorithm has gathered a great deal of interest over the last decade and has been used for a variety of applications (Hutchison and Hill 1997; Spall 1998, 2000; Gerencser et al. 2001; Gao and Reynolds 2007). As a result of the stochastic perturbation, the calculated gradient is also stochastic, however the expectation of the stochastic gradient is the true gradient (Gao and Reynolds 2007). So one would expect that the performance of the basic SPSA algorithm to be similar to the performance of steepest descent.

The gradient-based algorithms are faster to converge than any objective function-based gradient approximations such as SPSA algorithm when speed is measure in terms of the number of iterations. The total cost to achieve effective convergence depends not only on the number of iterations required, but also on the cost needed to perform these iterations, which is typically greater in gradient-based algorithms. This cost may include greater computational burden and resources, additional human effort required for determining and coding gradients.

Vermeulen and Heemink (2006) proposed a method based on proper orthogonal decomposition (POD) which shifts the minimization into lower dimensional space and avoids the implementation of the adjoint of the tangent linear approximation of the original nonlinear model. Recently, Altaf et al. (2011) applied this POD-based calibration method for the estimation of depth values and bottom friction coefficients for a very large-scale tidal model. The method has also been applied in petroleum engineering by Kaleta et al. (2011) for history matching problems. One drawback of the POD-based calibration method is its dependence on the number of parameters.

In this paper the SPSA algorithm is applied for the estimation of depth values in the tidal model DCSM of the entire European continental shelf. A number of calibration experiments is performed both simulated and real data. The effectiveness of the algorithm is evaluated in terms of the accuracy of the final results as well as the computational costs required to produce these results. In doing so, comparison is made with a traditional steepest descent method and also with a newly developed POD-based calibration method.

The paper is organized as follows. Section 2 describes the SPSA algorithm. This section also briefly discusses the POD-based calibration approach which is used here as comparison with SPSA method. The following

section briefly explains the DCSM model used in this study. Section 4 contains results from experiments with the model DCSM, to estimate the water depth. The paper concludes in Section 5 by discussing the results.

## 2 Parameter estimation using SPSA

Consider a data assimilation problem for a general nonlinear dynamical system. The discrete system equation for the state vectors $X(t_{i+1}) \in \Re^n$ is given by;

$$X(t_{i+1}) = M_i[X(t_i), \gamma], \tag{1}$$

where $M_i$ is nonlinear and deterministic dynamics operator that includes inputs and propagates the state from time $t_i$ to time $t_{i+1}$, $\gamma$ is vector of uncertain parameters which needs to be determined. Suppose now that we have imperfect observations $Y(t_i) \in \Re^q$ of the dynamical system (1) that are related to model state at time $t_i$ through

$$Y(t_i) = HX(t_i) + \eta(t_i), \tag{2}$$

where $H : \Re^n \to \Re^{n^q}$ is linear observation operator that maps the model fields on observation space and $\eta(t_i)$ is unbiased random Gaussian error vector with covariance matrix $R_i$.

We assume that the difference between data and simulation results is only due to measurement errors and incorrectly prescribed model parameters. The problem of the estimation is then solved by directly minimizing the objective function $J$

$$J(\gamma) = \sum_i [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] \tag{3}$$

with respect to the parameters $\gamma$ satisfying the discrete nonlinear forecast model (1).

In the SPSA algorithm, we minimize the objective function $J(\gamma)$ using the iteration procedure

$$\gamma^{l+1} = \gamma^l - a_l \hat{g}_l(\gamma^l), \tag{4}$$

where $\hat{g}_l(\gamma^l)$ is a stochastic approximation of $\nabla J(\gamma^l)$, which denotes the gradient of the objective function with respect to $\gamma$ evaluated at the old iterate, $\gamma^l$. if $\hat{g}_l(\gamma^l)$ is replaced by $\nabla J(\gamma^l)$, then Eq. 4 represents the steepest descent algorithm.

The stochastic gradient $\hat{g}_l(\gamma^l)$ is SPSA algorithm is calculated by the following procedure.

1.  Define the $n^p$ dimensional column vector $\triangle_l$ by

$$\triangle_l = [\triangle_{l,1}, \triangle_{l,2}, \cdots, \triangle_{l,n^p}]^T, \tag{5}$$

and

$$\triangle_l^{-1} = [\triangle_{l,1}^{-1}, \triangle_{l,2}^{-1}, \cdots, \triangle_{l,n^p}^{-1}]^T, \tag{6}$$

where $\triangle_{l,i}, i = 1, 2, \cdots, n^p$ represents independent samples from the symmetric $\pm 1$ Bernoulli distribution. This means that $+1$ or $-1$ are the only possible values that can be obtained for each $\triangle_{l,i}$. It also means that

$$\triangle_{l,i}^{-1} = \triangle_{l,i}, \tag{7}$$

and

$$E[\triangle_{l,i}^{-1}] = E[\triangle_{l,1}] = 0, \tag{8}$$

where $E$ denotes the expectation.

2.  Define a positive coefficient $c_l$ and obtain two evaluations of the objective function $J(\gamma)$ based on the simultaneous perturbation around the current $\gamma^l$: $J(\gamma^l + c_l \triangle_l)$ and $J(\gamma^l - c_l \triangle_l)$.

3.  A realization of the stochastic gradient is then calculated by using central difference approximation as

$$\hat{g}_l(\gamma^l) = \frac{J(\gamma^l + c_l \triangle_l) - J(\gamma^l - c_l \triangle_l)}{2c_l} \triangle_l^{-1} \tag{9}$$

Since $\triangle_l$ is a random vector, $\hat{g}_l$ is also random vector. So by generating a sample of $\triangle_l$, we generate a specific sample of $\hat{g}_l$. The FDSA algorithm involves computation of each component of $\nabla J$ by perturbing one model parameter at a time. If one does a one-sided approximation for each partial derivative involved in $\nabla J(\gamma^l)$, then computation of the gradient requires $n^p + 1$ evaluations of $J$ for each iteration of the steepest descent algorithm. In contrast, the SPSA requires only two evaluations of the objective function $J(\gamma^l + c_l \triangle_l)$ and $J(\gamma^l + c_l \triangle_l)$ at each iteration.

### 2.1 Choice of $a_l$ and $c_l$

Returning to Eqs. 4 and 9, we see that we have left to specify with $a_l$ and $c_l$. These are specified here according to the guidelines given by Spall (1998). The relevant formulas for $a_l$ and $c_l$ are given by

$$a_l = \frac{a}{(A + l + 1)^{\hat{\alpha}}}, \tag{10}$$

and

$$c_l = \frac{c}{(l + 1)^{\hat{\beta}}}, \tag{11}$$

where a, c, A, $\hat{\alpha}$ and $\hat{\beta}$ are positive real numbers such that $0 < \hat{\alpha} \le 1$, $\hat{\alpha} - \hat{\beta} < 0.5$ and $\hat{\alpha} > 2\hat{\beta}$. The given choices for $\hat{\alpha}$, $\hat{\beta}$ will ensure that the algorithm,

Eq. 4 converges to a minimum of $J$ in a stochastic sense (almost surely). The choice of a, c, A, $\hat{\alpha}$ and $\hat{\beta}$ is to some extent case dependent and it may require some experimentation to determine good values of these parameters. Although the asymptotically optimal values of $\hat{\alpha}$ and $\hat{\beta}$ are 1.0 and 1/6, respectively (Chin 1997), but choosing smaller values, e.g., $\hat{\alpha} = 0.602$ and $\hat{\beta} = 0.101$ (Spall 1998) appear to be more effective in practice. One recommendation for A is to set A equal to 10% of the maximum number of iterations allowed.

The value of constant $c$ should be chosen so that $c$ is equal to the standard deviation of the noise in objective function $J$. If one has perfect objective function, then c should be chosen as small positive number.

## 2.2 Average stochastic gradient

One of the motivations for SPSA is that for a quadratic objective function such as $J$, the expectation of the stochastic gradient is the true gradient (Gao and Reynolds 2007), i.e.,

$$E[\hat{g}_l(\gamma^l)] = \overline{\hat{g}_l(\gamma^l)} = \nabla J(\gamma^l), \tag{12}$$

where $\overline{\hat{g}_l(\gamma^l)}$ is defined as

$$\overline{\hat{g}_l(\gamma^l)} = \frac{1}{N} \sum_{i=1}^{N} \hat{g}_l(\gamma^l), \tag{13}$$

with each $\hat{g}_l(\gamma^l)$ is obtained from Eq. 9 using $N$ different samples of $\triangle_l$. Due to the relationship given in Eq. 12, one would hope that SPSA would have convergence properties similar to those of steepest descent in terms of the number of iterations required to reduce the objective function $J$ to a certain level. In this case, SPSA could be much more efficient than the steepest descent algorithm.

## 2.3 POD-based calibration method

Vermeulen and Heemink (2006) proposed a method based on POD which shifts the minimization into lower dimensional space and avoids the implementation of the adjoint of the tangent linear approximation of the original nonlinear model. Due to the linear character of the POD-based reduced model its adjoint can be implemented easily and the minimization problem is solved completely in reduced space with very low computational cost.

The linearization of nonlinear high-order model (1) using the first order Taylor's formula around the background parameter $\gamma_k^b$ gives

$$\triangle \overline{X}(t_{i+1}) = \frac{\partial M_i[\mathbf{X}^b(t_i), \gamma^b]}{\partial X^b(t_i)} \triangle \overline{X}(t_i)$$
$$+ \sum_k \frac{\partial M_i[\mathbf{X}^b(t_i), \gamma^b]}{\partial \gamma_k} \Delta \gamma_k \tag{14}$$

where $\overline{X}$ is linearized state vector, $X^b$ is the background state vector with the prior estimated parameters vector $\gamma^b$ and $\triangle \overline{X}$ is a deviation of the model from background trajectory.

A model can be reduced if the incremental state $\triangle \overline{X}(t_{i+1})$ can be written as linear combination:

$$\triangle \overline{X}(t_i) = P\xi(t_{i+1}) \tag{15}$$

where $P = \{p_1, p_2, \cdots, p_r\}$ is a projection matrix such that $P^T P = I$ and $\xi$ is a reduced state vector given by:

$$\begin{pmatrix} \xi(t_{i+1}) \\ \Delta \gamma \end{pmatrix} = \begin{pmatrix} \widetilde{M}_i & \widetilde{M}_i^\gamma \\ 0 & I \end{pmatrix} \begin{pmatrix} \xi(t_i) \\ \Delta \gamma \end{pmatrix} \tag{16}$$

Here, $\Delta \gamma$ is the control parameter vector, $\widetilde{M}_i$ and $\widetilde{M}_i^\gamma$ are simplified dynamics operators which approximate the full Jacobians $\frac{\partial M_i}{\partial X^b}$ and $\frac{\partial M_i}{\partial \gamma_k}$, respectively:

$$\widetilde{M}_i = P^T \frac{\partial M_i}{\partial X^b(t_i)} P \tag{17}$$

$$\widetilde{M}_i^\gamma = P^T \left( \frac{\partial M_i}{\partial \gamma_1}, \cdots, \frac{\partial M_i}{\partial \gamma_{n^p}} \right) \tag{18}$$

The dimension on which the reduced model operates is $(r + n^p) \times (r + n^p)$ with $n^p$ being the number of estimated parameters.

### 2.3.1 Collection of the snapshots and POD basis

The POD method is used here to obtain an approximate low-order formulation of the original tangent linear model. POD is an optimal technique of finding a basis which spans an ensemble of data (snapshots) collected from an experiment or a numerical simulation of a dynamical system. The reduced model used here is to estimate uncertain parameters, the snapshots should be able to represent the behavior of the system for these parameters. Therefore the snapshot vectors $e_i \in \Re^s$ are obtained from the perturbations $\frac{\partial M_i}{\partial \gamma_k}$ along each estimated parameter $\gamma_k$ to get a matrix

$$E = \{e_1, \cdots, e_s\}; i = \{1, 2, \cdots, s\}. \tag{19}$$

The dimension of this ensemble matrix $E$ is $s = n^p \times n^s$, where $n^s$ is the number of snapshot collected for each

individual parameter $\gamma_k$. The covariance matrix $Q$ can be constructed from the ensemble $E$ of the snapshots by taking the outer product

$$Q = EE^T \tag{20}$$

This covariance matrix is usually huge as in the current application with state vector of dimension $\sim 3 \times 10^6$, so direct solution of eigenvalue problem is not feasible. To shorten the calculation time necessary for solving the eigenvalue problem for this high-dimensional covariance matrix, we define a covariance matrix $G$ as an inner product

$$G = E^t E \tag{21}$$

In the method of snapshots (Sirovich 1987), one then solves the $s \times s$ eigenvalue problem

$$G\mathbf{z}_i = E^t E\mathbf{z}_i = \lambda_i \mathbf{z}_i, i \in \{1, 2, \cdots, s\} \tag{22}$$

where $\lambda_i$ are the eigenvalues of the above eigenvalue problem. The eigenvectors $\mathbf{z}_i$ may be chosen to be orthonormal and the POD modes $P$ are then given by:

$$\mathbf{p}_i = E\mathbf{z}_i/\sqrt{\lambda_i} \tag{23}$$

We define a measure $\psi_i$ for the relative information to choose a low dimensional basis by neglecting modes corresponding to the small eigenvalues:

$$\psi_i = \frac{\lambda_i}{\sum_{l=1}^{s} \lambda_l} 100\%, i = \{1, 2, \cdots, s\} \tag{24}$$

We collect $p_r$ $(r < s)$ modes such that $\psi_1 > \psi_2 > \ldots > \psi_r$ and they totally explain at least the required variance $\psi^e$,

$$\psi^e = \sum_{l=1}^{r} \psi_l \tag{25}$$

The total number of eigenmodes $r$ in the POD basis $P$ depends on the required accuracy of the reduced model.

*2.3.2 Approximate objective function and its adjoint*

In POD-based approach, we look for an optimal solution of Eq. 1 to minimize the approximate objective function ($\hat{J}$) in an incremental way:

$$\hat{J}(\Delta\gamma) = \sum_i [\{Y(t_i) - H(X^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)]^T$$
$$\times R_i^{-1} [\{Y(t_i) - H(X^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)] \tag{26}$$

The value of the approximate objective function $\hat{J}$ is obtained by correcting the observations $Y(t_i)$ for

background state $X^b(t_i)$ which is mapped on the observational space through a mapping $H$ and to the reduced model state $\xi(t_i, \Delta\gamma)$ which is mapped to the observational space through mapping $\hat{H}$, with $\hat{H} = HP$.

Since the reduced model has linear characteristics, it is easy to build an approximate adjoint model for the computation of gradient of the approximate objective function (26). The gradient of $\hat{J}$ with respect to $\Delta\gamma$ is given by:

$$\frac{\partial \hat{J}}{\partial(\Delta\gamma)} = \sum_i -[\hat{v}(t_{i+1})]^T \frac{\partial \xi(t_{i+1})}{\partial(\Delta\gamma)} \tag{27}$$

where $\hat{v}(t_{i+1})$ is the reduced adjoint state variable (Vermeulen and Heemink 2006). Once the gradient has been computed, the process of minimizing the approximate objective function $\hat{J}$ is done along the direction of the gradient vector in the reduced space.

Recently, Altaf et al. (2011) applied this POD-based calibration method for the estimation of depth values and bottom friction coefficients for a very large-scale tidal model. The method has also been recently applied in petroleum engineering by Kaleta et al. (2011) for history matching problems. One drawback of the POD-based calibration method is its dependence on the number of parameters.

## 3 The Dutch Continental Shelf Model

The DCSM is an operational storm surge model, used in the Netherlands for real-time storm surge prediction in North sea. Accurate predictions of the storm surges are of vital importance to the Netherlands since large areas of the land lie below sea level. Accurate forecasts at least six hours ahead are needed for proper closure of the movable storm surge barriers in Eastern Scheldt and the New Waterway. The governing equations used in DCSM are the nonlinear 2D shallow water equations. The shallow water equations, which describe large-scale water motions, are used to calculate the movements of the water in the area under consideration. These equations are

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + g\frac{\partial h}{\partial x} - fv + \frac{gu\sqrt{u^2 + v^2}}{HC_{2D}^2}$$
$$= \frac{1}{\rho_w}\frac{\tau_x}{H} - \frac{1}{\rho_w}\frac{\partial p_a}{\partial x}, \tag{28}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + g\frac{\partial h}{\partial y} + fu + \frac{gv\sqrt{u^2 + v^2}}{HC_{2D}^2}$$

$$= \frac{1}{\rho_w}\frac{\tau_y}{H} - \frac{1}{\rho_w}\frac{\partial p_a}{\partial y}, \qquad (29)$$

$$\frac{\partial h}{\partial t} + \frac{\partial Hu}{\partial x} + \frac{\partial Hv}{\partial y} = 0, \qquad (30)$$

where

$x, y$    Cartesian coordinates in horizontal plane
$t$    time coordinate
$u, v$    depth-averaged current in $x$ and $y$ direction, respectively
$h$    water level above reference plane
$D$    water depth below the reference plane
$H$    total water depth ($D + h$)
$f$    coefficient for the Coriolis force
$C_{2D}$    Chezy coefficient
$\tau_x, \tau_y$    wind stress in $x$ and $y$ direction, respectively
$\rho_w$    density of sea water
$p_a$    atmospheric pressure
$g$    acceleration of gravity

These equations are descretized using an alternating directions implicit (ADI) method and the staggered grid that is based on the method by Leendertse (1967) and improved by Stelling (1984). In the implementation, the spherical grid is used instead of rectangular (see e.g. Verboom et al. 1992). Boundary conditions are applied at both closed and open boundaries. At closed boundaries, the velocity normal to the boundary is zero. So no inflow and outflow can occur through these boundaries. At the open boundaries, the water level is described in terms of different harmonic components as follows:

$$h(t) = h_0 + \sum_{j=1}^{10} f_j H_j \cos(\omega_j t - \theta_j) \qquad (31)$$

where

$h_0$    mean water level
$H$    total water depth
$f_j H_j$    amplitude of harmonic constituent $j$
$\omega_j$    angular velocity of $j$
$\theta_j$    phase of $j$

All the open boundaries of the model are located in deep water (more than 200 m), see Fig. 1. This is done in order to explicitly model the nonlinearities of the surge tide interaction. A uniform initial water level of 0 m mean sea level has been used. For the initial velocity zero flow conditions have been prescribed. The time zone of the model is GMT.

### 3.1 Estimation of depth

The bathymetry for a model is usually from nautical maps. These maps usually give details of shallow rather than deep-water areas. If we use these maps to prescribe the water depth, it is reasonable to assume that this prescription of the bathymetry is erroneous. So depth can be a parameter on which model can be calibrated. In the early years of the developments of the DCSM, the changes to bathymetry were made manually. Later automated calibration procedures based on variational data assimilation were developed (Ten-Brummelhuis et al. 1993; Mouthaan et al. 1994). The complete description on the development of these calibrated procedures for DCSM can be found in Verlaan et al. (2005).
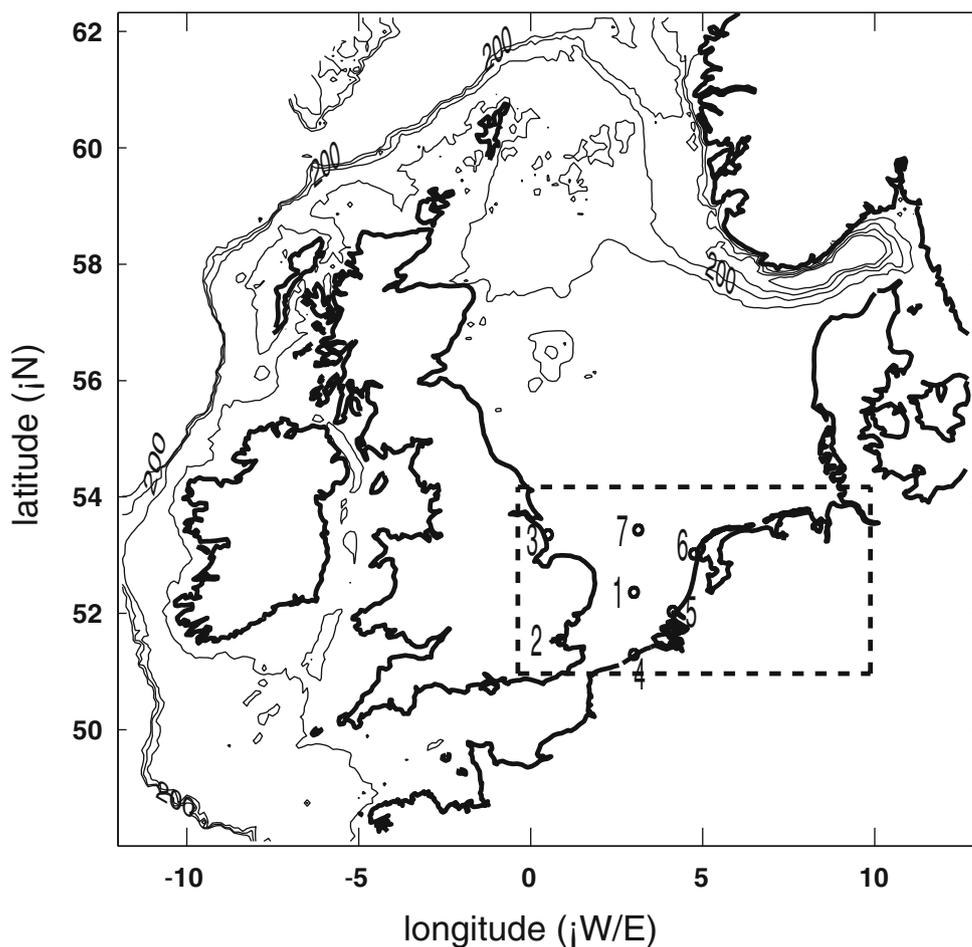
## 4 Numerical experiment

### 4.1 Experiment 1

The DCSM model used in this experiment covers an area in the north-east European continental shelf, i.e., 12°W to 13°E and 48°N to 62°N, as shown in Fig. 1. The resolution of the spherical grid is 1/8° × 1/12°, which is approximately 8 × 8 km. With this configuration there are 201 × 173 grid with 19,809 computational grid points. The time step is $\triangle t = 10$ min.

The depth values have to be prescribed at each grid cell of the model. Thus, theoretically it is possible to consider depth at each grid cell as a parameter to adapt. Practically it is not possible to take the adaptation values of every grid point as a parameter since far too many parameters would then have to be estimated in proportion to the available amount of data. Including too many parameters, identifiability will become a problem (Verlaan et al. 1996). This experiment was performed to assimilate data near the Dutch coast, i.e., domain $\Omega$ as shown in Fig. 1 (dashed rectangle). The rectangular areas were chosen, for which adaptation parameters were considered. These rectangular areas were chosen based on the previous calibrations of the DCSM (Ten-Brummelhuis 1992) and the spatial correlations within the rectangular regions. The numerical domain $\Omega$ was divided into seven subdomains $\Omega_k$, $k = 1, \cdots, 7$ see Fig. 2. For each subdomain $\Omega_k$, a correction parameters $\gamma_k^b$ was defined that was related to $D_{n_1,n_2}$ by:

$$D_{n_1,n_2} = D_{n_1,n_2}^b + \gamma_k^b; \textbf{if}(n_1, n_2) \in \Omega_k \qquad (32)$$

with $D_{n_1,n_2}^b$, the initial value. The parameters $\gamma_k^b$ were treated as unknown parameters. They acted as a correc-

tion for the mean level of the $D_{n_1,n_2}$ in a subdomain $\Omega_k$ and leave the spatial dependence inside $\Omega_k$ unaltered.

Seven observation points were included in the assimilation, two of which are located along the east coast of the UK, two along the Dutch coast and one at the Belgium coast (see Fig. 1). The truth model was run for a period of 15 days from 13 December 1997 00:00 to 27 December 1997 24:00 with the specification of water depth $D^b_{n_1,n_2}$ as used in the operational DCSM to generate artificial data at the assimilation stations. The first 2 days were used to properly initialize the simulations and set of observations $Y$ of computed water levels $h$ were collected for last 13 days at an interval of every ten minutes in seven selected assimilation grid points, which coincide with the points where data are observed in reality. The observations were assumed to be perfect. This assumption was made to see how close the estimate is to the truth; 5 m was added in $D^b_{n_1,n_2}$ at all the grid points in domain Ω to get the initial adjustments $\gamma^b_k$.

For the SPSA optimization algorithm, two methods were applied to calculate the stochastic gradient. In the first method, the stochastic gradient $\hat{g}_l(\gamma^l)$ was computed according to Eq. 9. In the second method, the gradient was computed by Eq. 13 referred as average
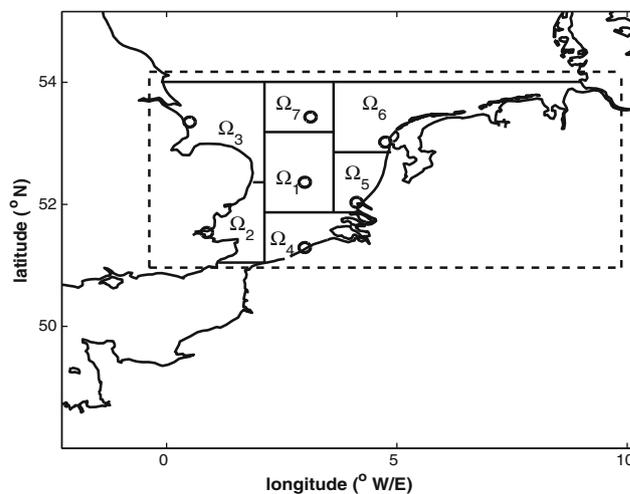


**Fig. 2** Shows the subdomains Ω1, Ω2, Ω3, Ω4, Ω5, Ω6, and Ω7

SPSA where expectation is taken over two independent stochastic gradients.

The values of a, c, A, $\hat{\alpha}$, and $\hat{\beta}$ were obtained according to the guidelines given in section 2.1. These values were determined as best from several forward model simulations. The iteration cycle for the SPSA algorithm was aborted when the value of the objective function $J$ did not change for the last three iterations of the minimization process (Wang et al. 2009).

Figure 3 shows a plot of the objective function $J$ versus number of iterations $\beta$ for the two implementations of the SPSA algorithms compared with the steepest descent and the POD-based calibration method. Note that the gradient used in the steepest descent algorithm is obtained from the finite difference method using one-sided perturbation. The graph shows that both SPSA and average SPSA gave comparable results, although for average SPSA the decrease in the objective function $J$ is more at early iterations. Also, the rate of convergence of average SPSA is slightly better than the SPSA. However, both SPSA and average SPSA are less efficient than steepest descent method. The steepest descent algorithm converged in ten iterations as compared to 20 and 15 iterations in SPSA and average SPSA, respectively. However, the cost of single iteration in SPSA algorithm is far less than the steepest descent algorithm.

For all the algorithms, there was a significant improvements in parameters for regions coinciding with the UK, Dutch and Belgian coast, but there was not much improvement in deep water regions $\Omega_1$ and $\Omega_7$. Since the subdomains containing deep areas are less sensitive as compared the subdomains containing shallow areas, so it is much difficult to estimate $\gamma_k$ in regions $\Omega_1$ and $\Omega_7$.
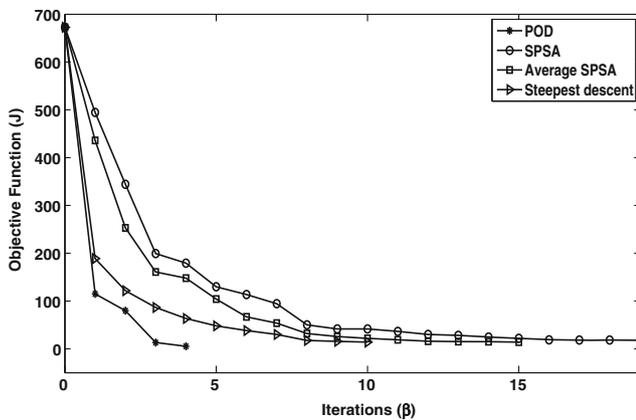
**Table 1** Comparison of estimated parameters to true parameters for the twin experiment

| $\zeta$ | SPSA (%) | Average SPSA (%) | Steepest descent (%) |
|---|---|---|---|
| All parameters | 35.11 | 29.27 | 21.02 |
| Sensitive parameters | 9.95 | 6.29 | 6.49 |

Table 1 lists the measure ($\zeta$) between the updated estimated parameters $\gamma^{up}$ obtained after calibration with different optimization algorithms and the true parameter estimate $\gamma^t$. The measure is defined as the two norm of the difference between estimated parameters $\gamma^{up}$ obtained after optimization and the true parameter estimate $\gamma^t$ divided by the norm of the true parameter estimate $\gamma^t$ (Gao and Reynolds 2007).

$$\zeta = \frac{\|\gamma^{up} - \gamma^t\|_2}{\|\gamma^t\|_2} \qquad (33)$$

By this measure, steepest descent (21%) performed the best followed by average SPSA (29%) and SPSA (35%). Since the stochastic gradient in the SPSA algorithm is based on two perturbations of the independent random samples, it is more likely that the SPSA algorithm improves more sensitive areas. The table also lists the same measure for shallow regions. In this case, all the algorithms steepest descent (6.49%), average SPSA (6.29%) and SPSA (9.95%) performed very well. Here, average SPSA matched the performance of the steepest descent algorithm. In average SPSA, the gradient was the average of only two independent stochastic gradients. One would expect better performance by the inclusion of more stochastic gradients in average SPSA. The Dutch continental shelf model (Table 2) presents the RMSE between estimated parameters ($\gamma^{up}$) and the true parameters ($\gamma^t$) after iterations $\beta = 5$, $\beta = 10$, $\beta = 15$ and $\beta = 20$ of SPSA algorithm for calibration stations and compares it with average SPSA and steepest descent algorithms. The RMSE for SPSA algorithm after iteration $\beta = 5$ is 9.95 compared to 8.92 and 6.05 in average SPSA and steepest descent algorithm, respectively. So SPSA and average SPSA are comparable at this point. The RMSE for SPSA after ten iterations is



**Fig. 3** Successive iterations $\beta$ of the minimization process

**Table 2** RMSE results for the minimization process after 5th, 10th, 15th, and 20th iterations

| | SPSA (cm) | Average SPSA (cm) | Steepest descent (cm) |
|---|---|---|---|
| Initial | 22.80 | 22.80 | 22.80 |
| $\beta = 5$ | 9.95 | 8.92 | 6.05 |
| $\beta = 10$ | 5.63 | 4.09 | 2.91 |
| $\beta = 15$ | 4.10 | 3.27 | – |
| $\beta = 20$ | 3.55 | – | – |

comparable to the RMSE of steepest descent method after only five iterations. Since the cost of one iteration of steepest descent is eight model simulations compared to two model simulations in SPSA algorithm, SPSA is two times more efficient than steepest descent at this point and one would expect SPSA to be more efficient if we have large number of parameters.

The RMSE with SPSA after $\beta = 15$ and average SPSA after $\beta = 10$ is similar. At this point the computational costs of both SPSA and average SPSA are also comparable. It is also clear from the Table 2 that the smallest RMSE value is achieved by steepest descent method in ten iterations.

Figure 4 presents water levels $h$ at the two tide gauge stations Den Helder and Southend along the Dutch and English coasts, respectively for the period from 18 December 1997 00:00 to 18 December 1997 24:00. These time series refer to water levels obtained from true values of the parameters, the initial values

of the parameters and the estimated values of the parameters using SPSA algorithm, respectively. Figure 4 demonstrates that the estimation methods significantly reduces the differences between time series obtained from initial parameters and the true parameters as compared with the differences between time series obtained from the estimated parameters and true parameters.

## 4.2 Experiment 2

The DCSM model used in this experiment is a newly designed spherical grid model. This newly developed DCSM covers an area in the north-east European continental shelf, i.e., 15° W to 13° E, and 43° to 64° N, as shown in Fig. 5. The spherical grid has a uniform cell size of 1/40° in east-west direction and 1/60° in north-south direction which corresponds to a grid cell size of about $\sim 2 \times 2$ km. With this configuration there are 1,120 grid cells in east-west direction and 1,260 grid

**Fig. 4** Water level time series for the period from 18 December 1997 00:00 to 18 December 1997 24:00 obtained from truth model, deterministic model with initial values of the estimated parameters and deterministic model after calibration, respectively, at the two tide gauge stations **a** Den Helder and **b** Southend
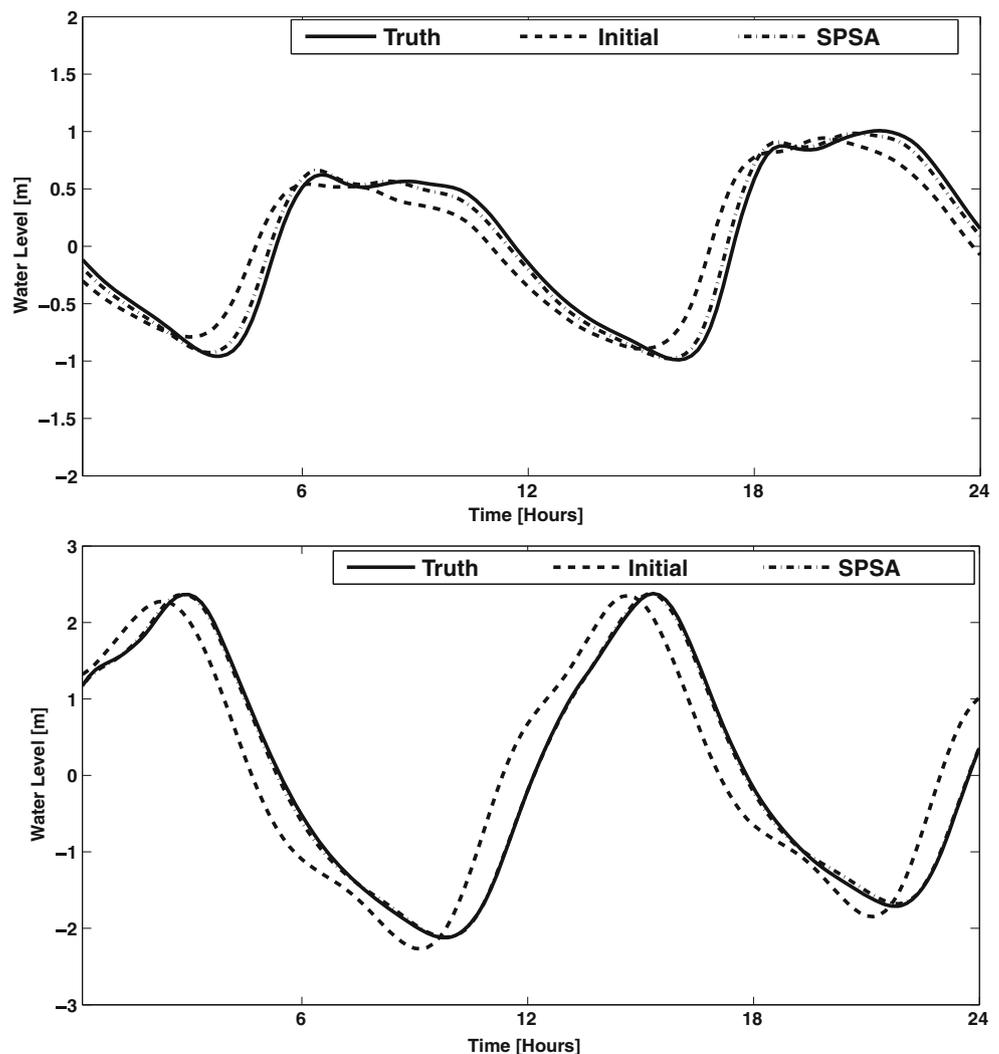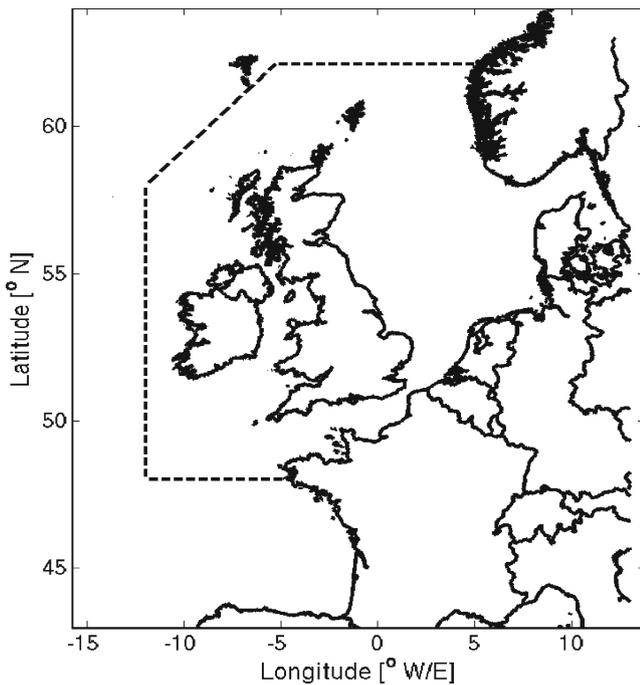
**Fig. 5** Newly developed hydrodynamic DCSM area. The *dashed line* represents the area of the operational DCSM extent



**Fig. 6** DCSM model bathymetry in meters. The bathymetry greater than 2,000 m is shown as 2,000 m

cells in north-south direction. The grid cells that include land are excluded form the model by the enclosures and the model contains 869,544 computational grid points. The grid resolution of the spherical grid is factor five finer then the DCSM model grid used in the previous experiment. The idea is to perform numerical experiment with a very large-scale model and with real data using SPSA algorithm.

The bathymetry of the model here is based on a NOOS gridded data set and for some areas in the model, ETOPO2 bathymetry data is interpolated on the computational grid (Ray 1999). The model bathymetry is presented in Fig. 6. The dashed line in Fig. 5 shows the comparison of the newly developed DCSM model area with the old DCSM. The model area of the newly developed DCSM is extended significantly in order to ensure that the open boundary conditions are located further away in deep water. A computational time step of 2 min has been applied. So to complete a 1 year model run on eight 3.6 MHz CPUs takes more than 2 days.

The model performance can be assessed by comparing it to the measured (observed) dataset. The available data used in this research consisted of two datasets of the tide gauge stations, namely,

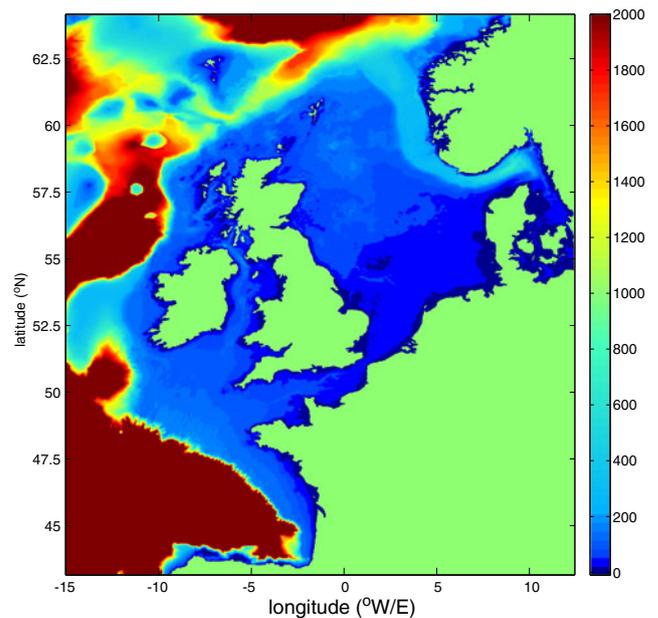1. water level measurement data from the Dutch DONAR database and

2. British Oceanographic Data Center offshore water level measurement data.

For the calibration, 50 water level locations are selected (see Fig. 7). Observations obtained by the harmonic analysis from these 50 stations at every fifth time step (10 min) were used for the calibration experiments. The calibration runs were performed for the period from 28 December 2006 to 30 January 2007 (34 days). The first 3 days were used to properly initialize the simulation. The measurement data were used for the remaining 30 days. This period was selected such that two spring neap-tide cycles are simulated. We have assumed that the observations $Y$ of the computed water levels $h$ contain an error described by white noise process with standard deviation $\sigma_m = 0.10$ (m).

The experiment was performed to estimate depth values using SPSA algorithm in this large-scale tidal model. The numerical domain $\Omega$ was divided into the 12 subdomains $\Omega_k$, $k = 1, \ldots, 12$ (see Fig. 8). The influence of the depth adjustments is quite significant specially in shallow regions. Thus, the subdivision of model area was made such that both deep and shallow areas were separated (see Fig. 8). The data observation points are concentrated in the English Channel, so this region was divided into five subdomains to improve the results by considering the local effects of the depth in each subdomain $\Omega_k$, $k = 3, \cdots, 7$, in this area.

Figure 9 shows a plot of the objective function $J$ versus number of iterations $\beta$ for the SPSA algorithm

**Fig. 7** DCSM area with stations included in the model calibration



compared with the POD-based calibration method. The SPSA method is compared here with POD-based calibration method for practical reasons. One reason is we have seen in the previous experiment that the POD-based calibration method efficiently estimated the depth values with the fastest convergence rate as compared to SPSA and steepest descent algorithms. Secondly, its not worthwhile to compute gradient by finite differences in this large-scale model. The graph shows that both the calibration methods give comparable results in terms of reduction in the objective function $J$. Though the rate of convergence of the POD-based calibration method is far better than the SPSA.

The POD-based calibration method converged in only two iterations as compared to 14 iterations with the SPSA, respectively. However, the cost of single iteration in the POD-based calibration method is much higher and is dependent on the number of parame-

ters $n^p$ and the POD modes $r$ used to construct the reduced model (Altaf et al. 2009). So for this experiment one iteration of the POD method required 13 initial simulations of the original nonlinear model to get the ensemble and then additional simulations of the original model to construct the POD reduced model in each iteration $\beta$ of the optimization process. The SPSA method on the other hand required only two objective function evaluations to compute the gradient in each iteration $\beta$ of the optimization procedure. For this application, the POD method is also fast since it is not needed to use a full simulations of the original model for the generation of the ensemble (Altaf et al. 2011). One disadvantage of POD-based calibration method is if the number of parameters is large the size of ensemble becomes large too and to construct a good reduced model is usually difficult with large ensemble size. For both the experiments performed the SPSA algorithm
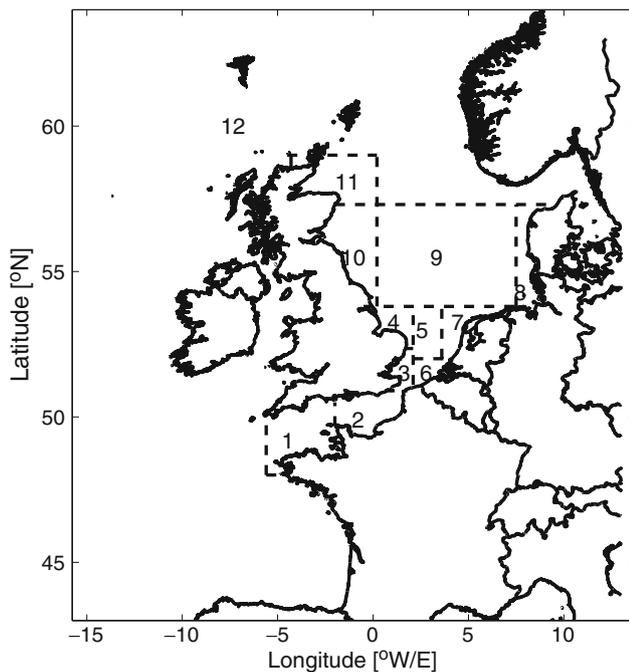
**Fig. 8** The 12 subdomains $\Omega_k$ of the DCSM used in Experiment 2

converged in almost similar iterations although the number of parameters were different. So, it is expected that the SPSA algorithm will work even with more parameters as the SPSA algorithm is independent of the number of the estimated parameters.
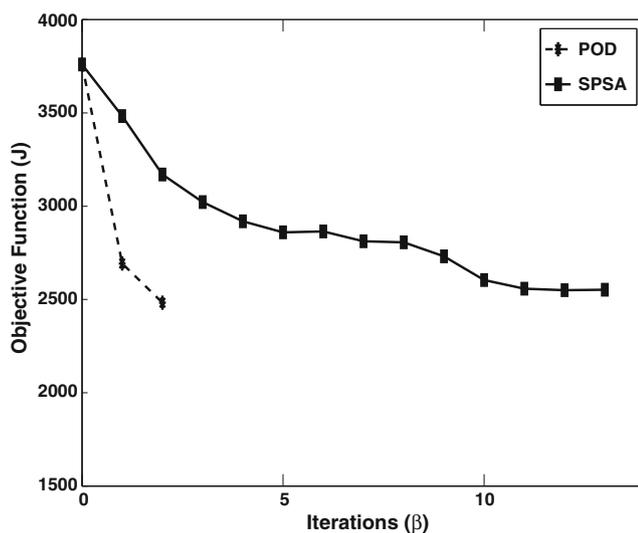


**Fig. 9** Successive iterations $\beta$ of the minimization process

## 5 Conclusions

In the absence of the adjoint model, the gradient is usually obtained by objective function evaluations to obtain the gradient approximations. Each individual model parameter is perturbed one at a time and the partial derivatives of the objective function with respect to the each parameter is estimated. This method is not feasible for automated calibration when large number of parameters are estimated. Simultaneous perturbation stochastic approximation (SPSA) method uses stochastic simultaneous perturbation of all model parameters to generate a search at each iteration. SPSA is based on a highly efficient and easily implemented simultaneous perturbation approximation to the gradient. This gradient approximation for the central difference method uses only two objective function evaluation independent of the number of parameters being optimized.

SPSA algorithm is applied to calibrate the model DCSM. The DCSM is an operational storm surge model, used in the Netherlands for real-time storm surge prediction in North sea. A number of calibration experiments was performed both with simulated and real data. The results from twin experiment showed that SPSA has a lower convergence rate than the steepest descent and POD-based calibration methods. The steepest descent algorithm converged in ten iterations as compared to 20 and 15 iterations in SPSA and average SPSA, respectively. However, the computational cost of single iteration in the steepest descent and the POD-based calibration methods is much higher and is dependent on the number of parameters $n^p$. Although both SPSA and steepest descent methods converged to similar value of the objective function, none of the optimization algorithms achieved the expected reduction in the objective function.

The results from a very large-scale tidal model and with real data showed that SPSA algorithm gives comparable results to POD-based calibration method. The POD-based calibration method converged in only two iterations as compared to 14 iterations with the SPSA, respectively. The POD-based calibration method though required 13 initial simulations of the original model to get the ensemble and then extra simulations to construct the POD reduced model in each iteration $\beta$ of the optimization process. The SPSA method on the other hand required only two objective function evaluations to compute an approximation of the gradient in each iteration $\beta$ of the optimization procedure independent of the number of estimated parameters. Thus, SPSA algorithm proved to be a promising optimization algorithm for model calibration for cases

where adjoint code is not available for computing the gradient of the objective function.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

## References

Altaf MU, Heemink AW, Verlaan M (2009) Inverse shallow-water flow modelling using model reduction. Int J Multiscale Com Eng 7:577–596

Altaf MU, Verlaan M, Heemink AW (2011) Efficient identification of uncertain parameters in a large scale tidal model of European continental shelf by proper orthogonal decomposition. Int J Numer Methods Fluids. doi:10.1002/fld.2511

Bennet AF, Mcintosh PC (1982) Open ocean modeling as an inverse problem: tidal theory. J Phys Oceanogr 12:1004–1018

Chin DC (1997) Comparative study of stochastic algorithms for system optimization based on gradient approximation. IEEE Trans Syst Man Cybern 27:244–249

Courtier P, Talagrand O (1990) Variational assimilation of meteorological observations with the direct and adjoint shallow water equations. Tellus 42:531

Courtier P, Thepaut JN, Hollingsworth A (1994) A strategy for operational implementation of 4d-var, using an incremental approach. Q J R Meteorol Soc 120:1367–1387

Gao G, Reynolds AC (2007) A stochastic algorithm for automatic history matching. SPE J 12:196–208

Gerencser L, Hill SD, Vagoo Z (2001) Discrete optimization via spsa. In: Proc. of American control conference, USA

Heemink AW, Mouthaan EEA, Roest MRT (2002) Inverse 3D shallow water flow modeling of the continental shelf. Cont Shelf Res 22:465–484

Hoteit I (2008) A reduced-order simulated annealing approach for four-dimensional variational data assimilation in meteorology and oceanography. Int J Numer Methods Fluids 58:1181–1199. doi:10.1002/fld.1794

Hutchison DW, Hill SD (1997) Simulation optimization of airline delay with constraints. In: Proc. 36th IEEE conference on decision and control, San Diego, USA

Kaleta MP, Henea RG, Jansen JD, Heemink AW (2011) Model-reduced gradient-based history matching. Comput Geosci 15:135–153

Kaminski T, Giering R, Scholze M (2003) An example of an automatic differentiation-based modeling system. Lect Notes Comput Sci 2668:5–104

Kiefer J, Wolfowitz J (1952) Stochastic estimation of a regression function. Ann Math Statist 23:462–466

Lardner RW, Al-Rabeh AH, Gunay N (1993) Optimal estimation of parameters for a two dimensional hydrodynamical model of the arabian gulf. J Geophys Res Oceans 98:229–242

Lawless AS, Nichols NC, Boess C, Bunse-Gerstner A (2008) Using model reduction methods within incremental 4dvar. Mon Weather Rev 136:1511–1522

Leendertse J (1967) Aspects of a computational model for long-period water wave propagation. Ph.D. thesis, Rand Corporation, Memorandom RM-5294-PR, Santa Monica

Mouthaan EEA, Heemink AW, Robaczewska KB (1994) Assimilation of ERS-1 altimeter data in a tidal model of the continental shelf. Dtsch Hydrogr Z 36(4):285–319

Navon IM (1998) Practical and theoratical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. Dyn Atmos Oceans (Special issue in honor of Richard Pfeffer) 27:55–79

Ray RD (1999) A global ocean tide model from topex/poseidon altimetry: Got99.2. NASA Technical Memorandum 209478

Seinfeld JH, Kravaris C (1982) Distributed parameter identification in geophysics-petroleum reservoirs and aquifers. In: Tzafestas, SG (ed) Distributed parameter control systems. Pergamon, Oxford. pp 367–390

Sirovich L (1987) Choatic dynamics of coherent structures. Physica D 37:126–145

Spall JC (1998) Implementation of the simultaneous perturbation algorithm for stochastic optimization. IEEE Trans Aerosp Electron Syst 34:817–823

Spall JC (2000) Adaptive stochastic approximation by the simultaneous perturbation method. IEEE Trans Automat Contr 45:1839–1853

Stelling GS (1984) On the construction of computational methods for shallow water flow problem. PhD thesis, Rijkswaterstaat Communications 35, Rijkswaterstaat

Ten-Brummelhuis PGJ (1992) Parameter estimation in tidal flow models with uncertain boundary conditions. Ph.D. thesis, Twente University, The Netherlands

Ten-Brummelhuis PGJ, Heemink AW, van den Boogard HFP (1993) Identification of shallow sea models. Int J Numer Methods Fluids 17:637–665

Thacker WC, Long RB (1988) Fitting models to inadequate data by enforcing spatial and temporal smoothness. J Geophys Res 93:10655–10664

Ulman DS, Wilson RE (1998) Model parameter estimation for data assimilation modeling: temporal and spatial variability of the bottom drag coefficient. J Geophys Res Oceans 103:5531–5549

Verboom GK, de Ronde JG, van Dijk RP (1992) A fine grid tidal flow and storm surge model of the north sea. Cont Shelf Res 12:213–233

Verlaan M, Mouthaan EEA, Kuijper EVL, Philippart ME (1996) Parameter estimation tools for shallow water flow models. Hydroinformatis 96:341–348

Verlaan M, Zijderveld A, Vries H, Kroos J (2005) Operational storm surge forcasting in the Netherlands: developments in last decade. Philos Trans R Soc A 363:1441–1453

Vermeulen PTM, Heemink AW (2006) Model-reduced variational data assimilation. Mon Weather Rev 134:2888–2899

Wang C, Gaoming L, Reynolds AC (2009) Production optimization in closed-loop reservoir management. SPE J 14:506–523