

PAPER • OPEN ACCESS

Fast automatic differentiation in problems variations four-dimensional data assimilation (4Dvar)

To cite this article: Y G Evtushenko *et al* 2018 *J. Phys.: Conf. Ser.* **1128** 012001

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Fast automatic differentiation in problems variations four-dimensional data assimilation (4Dvar)

Y G Evtushenko ¹, A Y Gorchakov ¹ and V M Goloviznin ^{2,3}

¹ Dorodnicyn Computing Centre, Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russia

² Nuclear Safety Institute of the Russian Academy of Sciences, Moscow, Russia

³ Lomonosov Moscow State University, Moscow, Russia

Abstract. The experience of using the technology of Fast Automatic Differentiation (FAD) is described to restore the initial data of a model hydrodynamic flow with a free boundary based on the results of spatial-temporal monitoring of velocity, density and free surface components. The solution of the direct problem is realized according to the CABARET scheme, for which the construction of the conjugate system is a serious independent task. Currently, there are available open computer codes that allow the implementation of FAD technology in the "black box" mode, submitting to its input the executable "author's" code for solving a direct problem. The model problem was solved with the help of the Adept 1.1 package, while using the 3-pass algorithm, which reduces the requirements for the amount of RAM.

Introduction

In the problems of atmospheric and ocean dynamics, the initial data are usually unknown and must be calculated as a result of the assimilation of monitoring data. One of the methods of assimilating such data is based on variational principles – the search for a minimum of some functional describing the deviation of the model solution and measurement data in some given metric. This method – 4Dvar (group of methods) is relatively well developed and successfully applied in practice (ECMWF, France, UK, Japan, Canada).

Numerical solution of such problems often uses ideas and methods developed earlier in studies on nonlinear programming. As the successful practice of the CC FRC CSC RAS has shown for many years, the solution of complex optimization problems, the most effective methods are gradient methods and their generalizations. For the calculation of gradients, the solution of a discrete problem conjugate to a "direct" problem, approximated by a difference or finite element system of equations, is usually used.

The "manual" derivation of the adjoint equations is very time-consuming and requires considerable time. Significant savings of resources are achieved using the methodology of fast automatic differentiation (FAD). In this case, the discrete gradient of the functional is determined by means of special canonical formulas [1]. There is no need to apply these formulas to the difference scheme manually. Currently, there are available open computer codes that allow the implementation of FAD technology in the "black box" mode, submitting to its input the executable "author's" code for solving a direct problem [2].

The paper is devoted to the experience of using FAD technology in solving the inverse problem of data assimilation for a model two-dimensional example, described by equations of single-layer shallow water with variable density over an uneven bottom. To solve the "direct" problem, the



balance-characteristic scheme "CABARET" was applied, since it has certain advantages over other schemes [3]. In addition, the computational template of the scheme is extremely compact, fitting into one space-time calculation cell. This allows setting the boundary conditions in a natural way in complex computational domains and providing the maximum computational efficiency of parallelization when performing massive parallel computations.

The subject of the study was the computational effectiveness of FAD in the problem of fluid dynamics with a free surface in a single-layer approximation, close to the tasks of operational oceanology. The conclusion is made about the prospects of using the CABARET scheme in combination with the FAD methodology for recovery of initial data.

1.Statement of the model problem

To test the effectiveness of FAD technology for operational oceanology, we consider a relatively simple model two-dimensional problem that contains the main features of this class of problems, namely, the problem of fluid dynamics with a free surface and variable density in the approximation of single-layer shallow water over an uneven bottom. Usually the shallow water equations are written out for the case of constant density, and, in this case, we consciously deal with the complexity of the task, in order to bring it closer to reality.

The system of differential equations describing the dynamics of single-layer shallow water with variable density follows from the laws of conservation of mass, momentum, and continuity equations and has the form:

$$\begin{aligned} \frac{\partial \rho h}{\partial t} + \frac{\partial \rho h u}{\partial x} + \frac{\partial \rho h v}{\partial y} &= 0; & \frac{\partial \rho h u}{\partial t} + \frac{\partial \rho h u^2}{\partial x} + \frac{\partial \rho h u v}{\partial y} + \frac{g}{2} \frac{\partial \rho h^2}{\partial x} &= -g \rho h \frac{\partial B}{\partial x}; \\ \frac{\partial \rho h v}{\partial t} + \frac{\partial \rho h v u}{\partial x} + \frac{\partial \rho h v^2}{\partial y} + \frac{g}{2} \frac{\partial \rho h^2}{\partial y} &= -g \rho h \frac{\partial B}{\partial y}; & \frac{\partial h}{\partial t} + \frac{\partial h u}{\partial x} + \frac{\partial h v}{\partial y} &= 0; \end{aligned} \tag{1}$$

Here ρ is the density, u, v are the velocity components along the directions x, y , respectively,

$B(x, y)$ is the bottom relief function, $h = H - B$ is the thickness of the liquid layer, and $H(x, y, t)$ is the level of the free surface. From the first and last equations of this system follows the density transport equation.

Let the area G in which the problem is considered have the form of a rectangle with sides L_x, L_y (Figure 1).

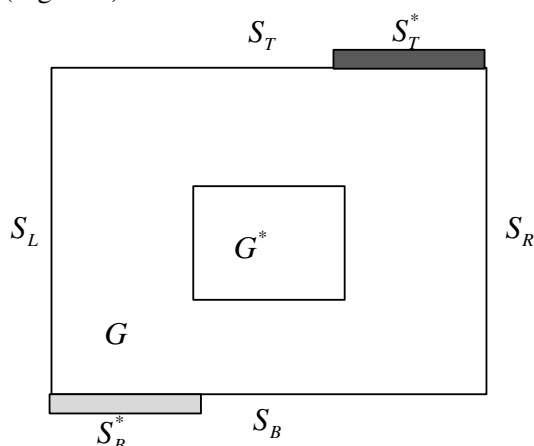


Figure 1. Area G

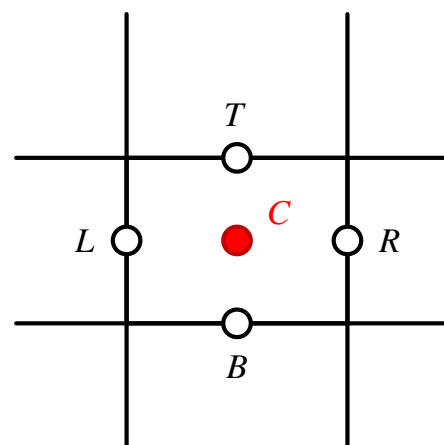


Figure 2. Scheme "CABARET"

The boundaries of this region will be denoted by S_R, S_L, S_T, S_B , which correspond to the right, left, upper and lower sides of the rectangle. The bottom relief function is chosen as a piecewise constant:

$$B(x, y) = \left\{ \Delta z, \quad \text{if } (x, y) \in G^* \text{ and } 0, \quad \text{if } (x, y) \in G / G^* \right\} \quad (2)$$

As the boundary conditions on the sides S_R, S_L , we set : $u|_{S_R} = u|_{S_L} = 0$.

On the part of the upper boundary S_T^* , we define the "upper reservoir" condition, and on the lower boundary part S_B^* - the "lower reservoir" condition, which determine the values of the Riemann invariants coming from outside the region:

$$\begin{aligned} H_T^* &= \text{const} > H(x, y, t_0); u_T^* = v_T^* = 0; \rho_T^* = f(t); (x, y) \in S_T^* \\ H_D^* &= \text{const} < H(x, y, t_0); u_B^* = v_B^* = 0; \rho_B^* = \rho_0; (x, y) \in S_B^* \end{aligned} \quad (3)$$

From the "upper reservoir" liquid flows into the region G , and from the region G it flows into the "lower reservoir". On the remaining sections of the upper and lower boundaries, the non-percolation condition is set: $v|_{S_T/S_T^*} = v|_{S_B/S_B^*} = 0$

We set the initial data as follows. We assume that at the initial moment t_0 the liquid is at rest, has a constant density, and its free surface is horizontal, and

$$H(x, y, t_0) = H_0; \quad H_T^* > H_0 > \Delta z; \quad H_0 > H_B^*; \quad \Delta z < H_B^*$$

The direct problem is solved until time t_1 at which the quasi-steady vortex flow can be considered as formed. In this time interval, the density of the inflowing liquid is considered equal to ρ_0 . Then, another density $\rho_1 > \rho_0$ is applied to the input and a dynamic substitution of one liquid for another takes place. We assume that the values of the initial parameters are given in such a way that the flow remains subsonic.

In the second time interval, a discrete spatial-temporal monitoring network Θ_m is specified that fixes the density, velocity components and the height of the free surface $\{\rho_k, u_k, v_k, H_k\}, k \in \Theta_m$.

The task of four-dimensional assimilation of monitoring data consists in determining the values of the density, velocity, and height of the free surface from these values at the instant of time t_1 .

2. An algorithm for numerical solution of a direct problem

For the numerical solution of a direct problem, we use the CABARET scheme, combining the merits of conservative and characteristic methods. The area G is covered by a regular grid with rectangular cells with sides $\Delta x, \Delta y$ (Figure 2). The calculated physical fields are referred to both the centers of the calculated cells (ρ_c, u_c, v_c, H_c) and to the centers of their faces (ρ_*, u_*, v_*, H_*) , where the asterisk takes on values $\{R, L, T, B\}$. The values relating to the centers of the cells are called "conservative", and to the midpoints of the faces – "fluxshape".

In the CABARET scheme three consecutive stages are distinguished (three phases). In the **first phase**, conservative variables are calculated in the intermediate time layer:

$$\frac{\bar{\psi}_c^{n+1/2} - \bar{\psi}_c^n}{\tau/2} + \bar{\Lambda}(\bar{\varphi}_f^n) = 0; \quad (4)$$

where $\bar{\psi}_c^n = \left\{ \left[(h), (\rho h), (\rho h u), (\rho h v) \right]_c^n \right\}^T$, $\bar{\Lambda}$ is the operator approximating the spatial derivatives in (1.1), the small index f indicates fluxshape variables on the faces of the cell. Conservative variables $\{h_c^{n+1/2}, \rho_c^{n+1/2}, u_c^{n+1/2}, v_c^{n+1/2}\}$ are evaluated sequentially explicitly.

Phase 2 calculates the values of fluxshape variables on a new time layer t_{n+1} . At this stage, the characteristic forms of equations (1.1) along the axis x are used:

$$\left\{ \frac{\partial u}{\partial t} + \frac{g}{c} \cdot \frac{\partial h}{\partial t} + \frac{c}{2\rho} \cdot \frac{\partial \rho}{\partial t} \right\} + (u+c) \cdot \left\{ \frac{\partial u}{\partial x} + \frac{g}{c} \cdot \frac{\partial h}{\partial x} + \frac{c}{2\rho} \cdot \frac{\partial \rho}{\partial x} \right\} = D_1^x;$$

$$\left\{ \frac{\partial u}{\partial t} - \frac{g}{c} \cdot \frac{\partial h}{\partial t} - \frac{c}{2\rho} \cdot \frac{\partial \rho}{\partial t} \right\} + (u-c) \cdot \left\{ \frac{\partial u}{\partial x} - \frac{g}{c} \cdot \frac{\partial h}{\partial x} - \frac{c}{2\rho} \cdot \frac{\partial \rho}{\partial x} \right\} = D_2^x; \tag{5}$$

$$\frac{\partial v}{\partial t} + u \cdot \frac{\partial v}{\partial x} = D_3^x; \quad \frac{\partial \rho}{\partial t} + u \cdot \frac{\partial \rho}{\partial x} = D_3^x;$$

A similar equation can be written along the axis y . Here $c = \sqrt{gh}$ is the speed of sound, the right-hand parts $D_k^x, D_k^y; k = 1, 2, 3, 4$ are fairly complex expressions that depend on the derivatives along the orthogonal direction. The form of these expressions in the CABARET scheme is not used.

It should be noted that the characteristic equations (5) in contrast to shallow water equations at constant density, can not be represented in total differentials and there are no Riemann invariants for the entire flow. However, when going from a differential problem to a discrete one for each computed space-time cell, you can enter the so-called "Local" Riemann invariants:

$$I_1^x = u + h \cdot \left(\frac{g}{c} \right)_c^{n+1/2} + \rho \cdot \left(\frac{c}{2\rho} \right)_c^{n+1/2}; \quad I_2^x = u - h \cdot \left(\frac{g}{c} \right)_c^{n+1/2} - \rho \cdot \left(\frac{c}{2\rho} \right)_c^{n+1/2}; \quad I_3^x = u; \quad I_4^x = \rho; \tag{6}$$

In terms of these expressions, the characteristic equations (1.8) are approximated as follows:

$$\frac{(I_k^x)_C^{n+1/2} - (I_k^x)_C^n}{\tau/2} + (\lambda_k)_C^{n+1/2} \cdot \frac{(I_k^x)_R^n + (I_k^x)_L^n}{\Delta x} = \{D_k^x\}_C^{n+1/2} + O(\tau, \Delta x^2); \quad k = 1, \dots, 4; \tag{7}$$

Similar formulas are obtained in the direction y . After the computation of phase 1, the conservative variables on the intermediate time layer are already known; therefore the left-hand sides of formulas (7) are also known and serve as a definition for the unknown right-hand sides.

Calculation of the values of local Riemann invariants on the new time layer is carried out by linear extrapolation. For the first two invariants in subsonic flows we obtain:

$$(I_1^x)_R^{n+1} = 2(I_1^x)_C^{n+1/2} - (I_1^x)_L^n; \quad (I_2^x)_L^{n+1} = 2(I_2^x)_C^{n+1/2} - (I_2^x)_R^n; \tag{8}$$

The direction of extrapolation for the third and fourth invariants depends on the direction of their transference speed in the intermediate time layer, which at each edge is assumed to be equal to half the sum of the conservative velocities in the adjacent cells:

$$\left. \begin{aligned} (I_3^x)_R^{n+1} &= (I_3^x)_C^{n+1/2} - (I_3^x)_L^n \\ (I_4^x)_R^{n+1} &= (I_4^x)_C^{n+1/2} - (I_4^x)_L^n \end{aligned} \right\} \text{if } u_R^{n+1/2} > 0; \quad \left. \begin{aligned} (I_3^x)_L^{n+1} &= (I_3^x)_C^{n+1/2} - (I_3^x)_R^n \\ (I_4^x)_L^{n+1} &= (I_4^x)_C^{n+1/2} - (I_4^x)_R^n \end{aligned} \right\} \text{if } u_L^{n+1/2} \leq 0; \tag{9}$$

Thus, four local invariants are calculated for the vertical faces of the calculation cells located within the computational domain. Similar formulas are obtained for horizontal faces T, B .

For the faces lying on the boundary of the domain, at best, three invariants will be computed. The missing invariant is replaced by the boundary condition - the vanishing of the normal component of the velocity vector on an impermeable wall, or by a local invariant coming outside the region from the "upper" or "lower" reservoir.

A very important operation in the CABARET scheme, as in all other high resolution schemes, is the non-linear correction of flows, which prevents the appearance of nonphysical oscillations in the solution. Here it is implemented on the basis of the maximum principle, which must obey local invariants - their values should be in the interval between the maximum and minimum, determined by the expressions:

$$\max(I_k^x) = \max\{I_L^x, I_C^x, I_R^x\}_k^n + \tau \cdot \{D_k^x\}_C^{n+1/2}; \quad \min(I_k^x) = \min\{I_L^x, I_C^x, I_R^x\}_k^n - \tau \cdot \{D_k^x\}_C^{n+1/2}; \quad k = 1, \dots, 4; \tag{10}$$

Nonlinear correction consists in assigning the nearest boundary value of the interval if the value of the local invariant goes beyond the interval defined in this way. After this, according to the corrected invariants, the values of the fluxshape variables on the new time layer are calculated.

The process of calculating one time step is completed by **phase 3**, where new values of conservative variables are found.

$$\frac{\bar{\psi}_C^{n+1} - \bar{\psi}_C^{n+1/2}}{\tau/2} + \bar{\Lambda}(\bar{\varphi}_f^{n+1}) = 0; \quad \bar{\psi}_C^n = \left\{ \left[(h), (\rho h), (\rho h u), (\rho h v) \right]_C^n \right\}^T \quad (11)$$

The CABARET difference scheme belongs to the class of high resolution schemes and has the second order of approximation on smooth solutions, when the flow correction procedure does not "work". It allows resolving thin vortex flow patterns on relatively coarse computation grids and is well parallelized. However, the "manual" construction of the conjugate system is a complex problem. Therefore, to calculate the gradient of the objective function, the methodology of fast automatic differentiation (FAD) was used [1].

3. Methodology of fast automatic differentiation

Let $z \in R^n, u \in R^r$ be vectors. The differentiable functions $W(z, u)$ and $\Phi(z, u)$ define mappings $W: R^n \times R^r \rightarrow R^1, \Phi: R^n \times R^r \rightarrow R^r$. The vectors z and u satisfy the following system of n nonlinear scalar equations $\Phi(z, u) = 0$. If the matrix $\Phi_z^T(z, u)$ is nondegenerate, then the complex function $\Omega(u) = W(z(u), u)$ is differentiable and its gradient with respect to the variables u is computed according to the formula:

$$\frac{d\Omega}{du} = W_u(z(u), u) + \Phi_u^T(z(u), u) \cdot p, \quad (12)$$

where the vector $p \in R^n$ is found from the solution of a linear algebraic system:

$$W_z(z, u) + \Phi_z^T(z, u) \cdot p = 0, \quad (13)$$

Here the index T denotes transposition, the subscripts z and u denote the partial derivatives of functions with respect to the vectors z and u : we also denote the i -th, j -th components of the vectors z and u as z_i, z_j, u_i, u_j .

Suppose that each component of vector z_i is sequentially expressed only through the components of vector u and the previous z_i , i.e. $1 \leq j < i$. Then formulas (12), (13) can be written in the following form:

$$p_i = \sum_{j=i+1}^n \Phi_{z_j}^T(z, u) p_j + W_{z_i}(z, u); \quad \frac{d\Omega}{du_i} = W_{u_i}(z, u) + \sum_{j=1}^n \Phi_{u_j}^T(z, u) \cdot p_j \quad (14)$$

In [1], the time complexity of calculating the gradient is estimated at $T_g/T_0 \leq 3$, where T_0 is the total time required to calculate the value of the function, and T_g is the additional time required to calculate all the partial derivatives of the function. The estimate is given without taking into account the time required to work with the computer's memory. Differentiation according to the formulas is implemented in many software packages FAD. This task was solved with the help of the package Adept 1.1 [2], while using a 3-pass algorithm [4], which reduces the requirements for the amount of RAM.

4. The numerical experiment

To perform the calculations, the computing resources of FIC IU RAS were used [5] - 2xIBM Power8 3.5 Hz, 512 GB RAM - a 2-processor server, each processor contains 8 cores supporting SMT8 technology (8 computational threads per core) - a total of 128 logical cores. The gcc 7.2.0 compiler was used. The area was covered with a regular grid 100x100x1000: 100 cells along the x axis, 100 cells along the y axis and 1000 cells along the time axis. The height of the liquid in the "upper"

reservoir $H_T^* = 1.1$, the height of the liquid in the "lower" reservoir $H_B^* = 0.9$, the density of the inflowing liquid $\rho_0 = 1$ and $\rho_1 = 1.1$. Parameters of the bottom relief function $B(x, y) - \Delta z = 0.3$ and G^* - a rectangle located in the center of the area G with sides $0.4L_x, 0.4L_y$ (see Figure 1).

The discrete space-time monitoring network Θ_m consisted of 1000 points randomly located (at each time, density, velocity components and free surface height were fixed at one point). The pattern of the vortex flow in the second time interval (the height of the free surface H) is shown in Figure 3.

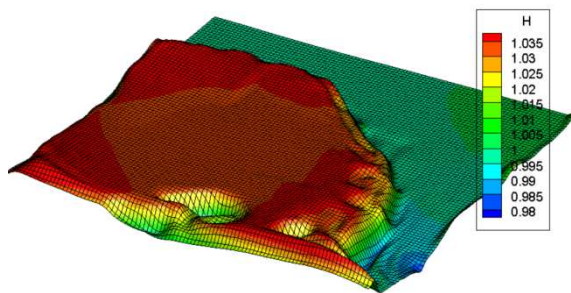


Figure 3. Height of the free surface H .

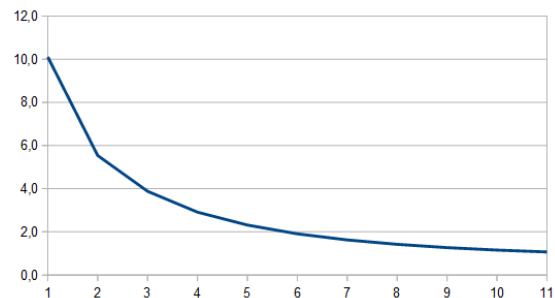


Figure 4. Convergence of the gradient descent method.

The difference scheme CABARET and the minimized functional were implemented in the C++ language, and the 3-pass FAD algorithm was used to calculate the gradient [4].

As a method of optimization, the gradient descent method was used. In the process of calculation, the memory consumption did not exceed 3GB (for comparison, the 2-pass algorithm in this problem required more than 400 GB of RAM). The convergence of the method with respect to the functional W for the first 10 steps is shown in Figure 4.

Conclusion

The application of the CABARET scheme, providing highly efficient parallel calculations on coarse computational grids and the FAD methodology, in particular a 3-pass gradient calculation algorithm will allow solving practical problems of operational oceanology. Further development of this approach involves the use of optimization methods with a higher rate of convergence, for example, Limited-memory BFGS in the family of quasi-Newton methods that approximates the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm or the Levenberg-Marquardt method.

Acknowledgments The authors express their gratitude to the undergraduates Pavel Mayorov and Petr Mayorov for their assistance in the calculations. The work was supported by the RSCF, project No. 18-11-00163.

References

- [1] Evtushenko Y G 1998 *Optim. Methods Software* **9** 45-75
- [2] Hogan R J 2014 *Fast reverse-mode automatic differentiation using expression templates in C++* (ACM Trans. Math. Softw. 40, 4, Article 26)
- [3] Goloviznin V M, Zaytsev M A, Karabasov S A and Korotkin I A 2013 *New algorithms of computational hydrodynamics for multiprocessor computer systems* (Moscow: Moscow University)
- [4] Gorchakov A Y 2017 *Int. J.l of Open Information Tech.* **12** 49-53
- [5] Federal Research Center "Computer Science and Control" of Russian Academy of Sciences [Electronic resource]: site. – (Moscow: FRC CSC RAS) - URL: <http://frcsc.ru>