Two-way embedding algorithms: a review

Laurent Debreu, Eric Blayo

▶ **To cite this version:**

Laurent Debreu, Eric Blayo. Two-way embedding algorithms: a review. Ocean Dynamics, 2008, 58 (5-6), pp.415-428. 10.1007/s10236-008-0150-9 . hal-00387449

HAL Id: hal-00387449
https://hal.science/hal-00387449v1

Submitted on 17 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Two-way embedding algorithms: a review

submitted to Ocean Dynamics: Special Issue on Multi-Scale Modelling: Nested Grid and Unstructured Mesh Approaches

Laurent Debreu · Eric Blayo

Abstract Local mesh refinement features have now been added to a number of numerical ocean models. In its crudest form, a high resolution grid is embedded (or nested) in a coarse resolution grid, which covers the entire domain, and the two grids interact. The aim of this paper is to review existing two way grid embedding algorithms. The basic algorithms and specificities related to ocean modelling are first described. Then we address several important issues: conservation properties, design of interpolation/restriction operators and noise control techniques.

Keywords Two way embedding · Mesh refinement · Structured grids

1 Introduction

An increase of the horizontal resolution of a numerical ocean model still remains a key point in the improvement of the realism of its solutions, mainly through a better representation of small scales and domain geometry. This increase of resolution is generally not performed everywhere in the domain, both because it is not necessary from a physical point of view and because of limited computational resources. That is why a number of ocean models include local mesh refinement features. The idea is to refine the mesh where (and potentially when) necessary according to the objective of the simulation.

For models based on a structured grid, a possible way to locally increase the resolution is to use a grid with variable resolution. Starting from a uniform grid, a mapping is introduced to produce a grid with increased resolution in areas of interest (Zhuo and Qingcun 1995). This approach has the advantage that the model is still written on a single grid and thus there is almost no additional coding complexity and properties like conservation are easily handled. The disadvantages lie mainly in the difficulty of

Laurent Debreu
INRIA, Laboratoire Jean Kuntzmann, 51 rue des Mathématiques, 38400 Saint Martin d'Hères, France, E-mail: Laurent.Debreu@inria.fr

Eric Blayo
Joseph Fourier University, Laboratoire Jean Kuntzmann

generating grids with good numerical properties, and in the fact that the time step constraint relies on the smallest grid cell (in the case of explicit time stepping schemes).
The second approach, which is the subject of the present paper, is to embed a high resolution (HR) version of the same model at a specific location. The information is exchanged in two ways between the coarse resolution (CR) solution and the HR solution: the coarse grid provides the boundary conditions of the child grid, while the high resolution solution is used to update the coarse grid solution in the common domain. The major interest of this approach is that the coarse grid provides accurate boundary conditions for incoming information along the HR grid boundaries while the update (or feedback) step enables outgoing information to leave the fine grid much better than in one way interaction (i.e. without feedback from the HR to the CR solution). In this approach, even if the same model is applied, numerical schemes and/or physical parameterizations can be adapted to the grid resolution.
We will discuss here only refinement of the mesh along the horizontal directions. A few applications of vertical mesh refinement have also been reported. In Fox and Maskell (1995), the vertical refinement is applied to the whole water column. For similar reasons to the non decomposition of the vertical direction in parallel applications (use of implicit schemes and particular treatment of the barotropic mode), local mesh refinement on the vertical introduces additional difficulties, some of them very close to the ones reported in this paper for horizontal refinement.
Concerning horizontal refinement, we will focus here on several important aspects including conservation, intergrid transfer operators and noise control techniques. We begin in section 2 with a description of the basic algorithms underlying the two way mesh refinement method and then give an overview of difficulties that can arise in practical numerical simulations. Conservation issues are addressed in section 3. In section 4, intergrid transfer operators (interpolations and updates) are examined and important details of implementations are given. Noise control techniques, which aim at making the algorithm more robust, are presented in section 5.

2 Two-Way nesting: Algorithms

In this paper, we will consider only one high resolution grid embedded in the coarse resolution grid. Extension to more than one grid and/or more than one level of refinement does not present additional complexities.
So let's consider Ω the domain covered by the coarse grid and a subdomain ω, covered by the fine grid. γ is the boundary of ω. At the discrete level, the numerical grids are denoted by Ω_H and ω_h (cf. Fig. 1). ω_H is the part of the grid Ω_H corresponding to the domain ω.

2.1 Basic Algorithm

The ratio between the coarse and fine horizontal mesh sizes is an integer ρ, the mesh refinement factor ($\rho = 2$ on Fig. 1). Typical values of ρ in actual applications range from 2 to 5. Using larger values introduces a too strong change in the resolved scales on the different grids. In that case, it may not be appropriate to try to enforce a strong coupling between the two grids and methods with weaker interactions than the ones we will discuss in this paper could be of interest (Sheng et al. 2005).

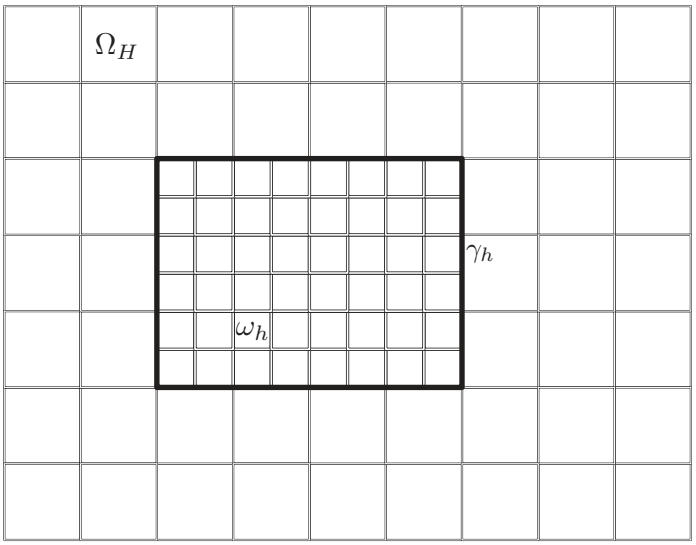


Fig. 1 Local mesh refinement of a structured grid with a mesh refinement factor of 2. The high resolution grid ω_h is embedded in the coarse resolution grid Ω_H.

Choosing an odd mesh refinement factor simplifies grid interactions since in that case a coarse grid point has always one underlying high resolution point. Figure 2 shows, for B and C grids (using the Arakawa classification), that if a variable is staggered (cell or face centered) and if the mesh refinement factor is even, then a coarse grid point does not have a corresponding point on the fine grid.

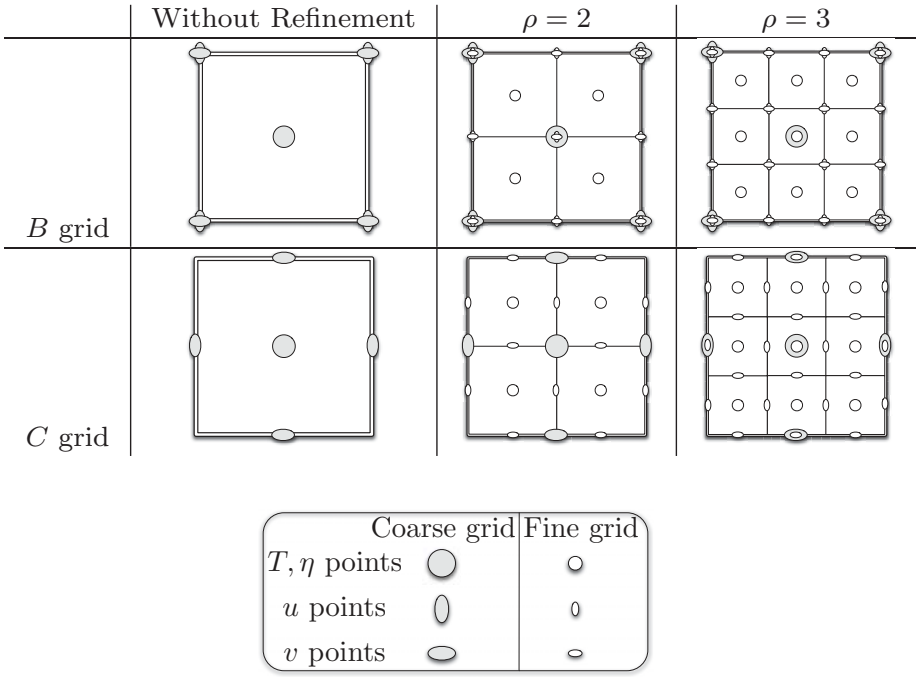


Fig. 2 Refinement of a cell with even ($\rho = 2$) and odd ($\rho = 3$) refinement factors for a B and a C grid. Only one coarse cell is shown. When $\rho = 3$, a coarse grid point has always a corresponding fine grid point.

Associated with this spatial mesh refinement, time refinement may also be applied. Most of the ocean models now use explicit time integration algorithms both for computational reasons (efficiency on distributed memory parallel computers) and for ac-

curacy reasons (smaller dispersion errors than implicit schemes). The model is then subject to a given CFL (Courant Friedrichs Lewy) stability condition and the ratio $\Delta t/\Delta x$ must be kept smaller than a given value on the whole grid hierarchy. As we shall see, refining in time induces several difficulties but is required for some applications, especially when only a small portion of the coarse grid domain is refined or when there are several levels of refinement. In these cases, keeping the same time step over the whole grid hierarchy greatly increases the computational cost.
The integration algorithm for a time refinement factor ρ_t of 3 is depicted on Fig. 3.

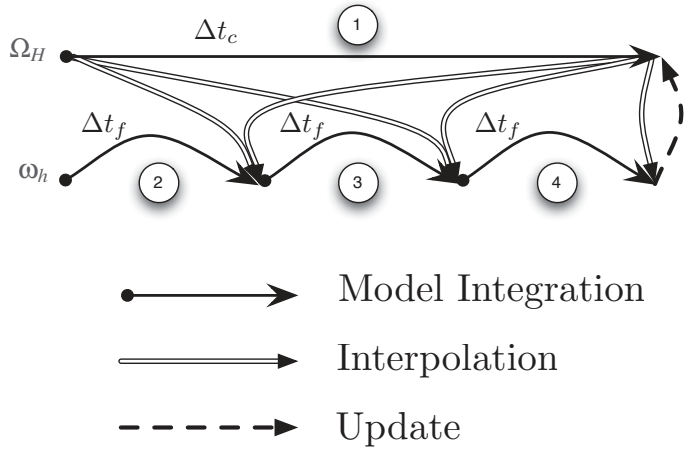


Fig. 3 Integration algorithm for a time refinement factor of 3

The model is first integrated on the coarse grid with a time step equal to Δt_c and then the fine grid is integrated ρ_t times with $\Delta t_f = \Delta t_c/\rho_t$. Interpolations of coarse grid boundary data occur at the end of each fine grid time step and updates occur at the end of coarse time step, when the two solutions have been advanced to the same physical time.
Let $\mathcal{L}$ represent the model integration from time t^n to time t^{n+1} and $\mathcal{L}_c, \mathcal{L}_f$ its discretizations on the CR and HR grids. Let P denote the interpolation (or prolongation) operator from Ω_H to γ_h, the boundary of ω_h, and R the update (or restriction) operator from ω_h to ω_H. Then, assuming that the model is fully explicit, the algorithm can be written in the following simplified form :

1. $u_c^{n+1} = \mathcal{L}_c(u_c^n)$
2. `For` $m = 1 \dots \rho_t$ `do`
   $$u_f^{n+\frac{m}{\rho_t}} = \mathcal{L}_f\left(u_f^{n+\frac{(m-1)}{\rho_t}}\right)$$
   $$u_f^{n+\frac{m}{\rho_t}}{}_{|\gamma_h} = P(u_c^n, u_c^{n+1})$$
3. $u_c^{n+1}{}_{|\omega_H} = R(u_f^{n+1})$

P, the interpolation operator, includes a time interpolation. This time interpolation is done linearly in most of the applications and thus requires only the knowledge of u_c at time t^n and t^{n+1}. To our knowledge, the effect of increasing the interpolation order in time has not been studied in the literature. This is because, as for numerical models on a uniform grid, the error relative to time is expected to be lower than the one relative to spatial interpolation. However, as for numerical models, it may not be the case anymore due to the increase of horizontal resolution and to the use of higher

order spatial schemes. New experiments are probably required to test the validity of the linear interpolation.
Note finally that interpolation and restrictions can also be applied to time tendencies instead of instantaneous values (Jones 1977; Spall and Holland 1991; Oey and Chen 1992).

2.2 Complexities associated with time stepping

The numerical schemes are seldom as simple as previously described and this can lead to several complications. In a numerical ocean model, some of them are linked to the particular treatment of the gravity waves associated with the evolution of the barotropic mode. This treatment leads either to the use an implicit time stepping, thus leading to the solution of an elliptic system, or to the use of a time splitting method.

2.2.1 Elliptic systems

Elliptic systems can come from a 2D implicit solve (e.g. implicit treatment of free surface (Dukowicz and Smith 1994) or rigid lid approximation). A 3D system has also to be solved if the hydrostatic assumption is removed. If we denote the associated linear system by :

$$\mathcal{A}v = \mathcal{B}$$

then after discretization, on the nested grid, the simplest choice is to force the high resolution solution with an interpolation of the coarse grid solution :

$$a)\ \mathcal{A}_c v_c = \mathcal{B}_c \text{ on } \Omega_H, \qquad b)\ \begin{cases} \mathcal{A}_f v_f = \mathcal{B}_f \text{ on } \omega_h \\ v_{f_{|\gamma_h}} = P v_c \end{cases} \tag{1}$$

This implies continuity of the solution at the interface γ_h but the resulting gradient across the boundary is discontinuous. Additionally a loss of accuracy is introduced because errors produced by the coarse grid resolution a) propagate inside the HR domain through the boundary forcing in step b).
A first step toward better accuracy is to update the right hand side of the coarse grid equation so that Eq. 1 is replaced by Eq. 2:

$$a)\ \mathcal{A}_c v_c = \begin{cases} R\mathcal{B}_f \text{ in } \omega_H \\ \mathcal{B}_c \quad \text{in } \Omega_H \backslash \omega_H \end{cases}, \qquad b)\ \begin{cases} \mathcal{A}_f v_f = \mathcal{B}_f \text{ on } \omega_h \\ v_{f_{|\gamma_h}} = P v_c \end{cases} \tag{2}$$

In Spall and Holland (1991), this technique is applied to a rigid lid ocean model. The linear system corresponds to the computation of the streamfunction tendency and the right hand side is the barotropic vorticity tendency. The barotropic vorticity tendency on the coarse grid is updated by the one computed on the fine grid.
However, despite this improvement, the gradient of the solution across the interface is still discontinuous. To be properly solved the two grid system should be handled as a truly local multilevel system. This can be done using local defect correction methods (see Laugier et al. 1996 for an application to a rigid lid model and a comparison with results obtained in Spall and Holland 1991). The system can also be seen as discretized

on a single composite grid. In this approach, the operator $\mathcal{A}$ is modified at the boundary and involves both coarse and fine grids variables. The resulting system is:

$$\begin{pmatrix} \mathcal{A}_c & 0 \\ 0 & \mathcal{A}_f \\ \mathcal{A}_{c\gamma} & \mathcal{A}_{f\gamma} \end{pmatrix} \begin{pmatrix} v_{c|_{\Omega_H \backslash \omega_H}} \\ v_f \end{pmatrix} = \begin{cases} \mathcal{B}_c \text{ in } \Omega_H \backslash \omega_H \\ \mathcal{B}_f \text{ on } \omega_h \\ \mathcal{B}_\gamma \text{ in } \gamma_h \end{cases} \tag{3}$$

$\mathcal{A}_{c\gamma} v_{c|_{\Omega_H \backslash \omega_H}} + \mathcal{A}_{f\gamma} v_f = \mathcal{B}_\gamma$ is an approximation of the original system on the interface which leads at convergence to continuity of both the solution and its gradient across the boundary. See Martin and Cartwright (1996) for applications of the local multilevel method to the solution of a Poisson equation.

Remarks:

- Algorithms 2 and 3 are more difficult to apply when time refinement is used. Indeed the right hand side $\mathcal{B}_c$ and $\mathcal{B}_f$ should be computed at the same time and the way to write the system at intermediate time steps on the fine grids is not obvious.
- Since we are using two way techniques, the right hand side $\mathcal{B}_c$, which depends on coarse grid variables that have been updated, is never significantly different from $R\mathcal{B}_f$. In some cases, the update schemes used for the primary variables are precisely chosen in such a way that $\mathcal{B}_c = R\mathcal{B}_f$ in ω_H. However, this leads most of the time to the use of very simple restriction operators (e.g. average, see Clark and Farley 1984) that do not have good filtering properties and thus can affect the quality of the solution (see paragraph 4.1).

2.2.2 Free surface and Time splitting

Similar problems are also inherent when a time-splitting is used for the treatment of the external mode. A number of models now use the time-splitting approach of Blumberg and Mellor (1987); Killworth et al. (1991). The barotropic time step is set to a fraction of the baroclinic time step. The barotropic mode is then integrated as a solution of a shallow water model forced by the vertical integral of the right hand side of the 3D equations. At the end of this integration, the newly computed barotropic quantities are used to replace the barotropic part of the 3D fields. In order to prevent aliasing errors, two approaches can be taken. The first one, as in Killworth et al. (1991), is to use in the barotropic time steps a time integration scheme that sufficiently damps high temporal frequencies (e.g. Euler Backward). In this case, the fine and coarse grid can interact during the small barotropic time steps throughout the barotropic quantities (see Oey and Chen 1992). This is illustrated on Fig. 4 with a time refinement factor of 2.
The second approach consists in applying a filter to the barotropic quantities to filter out the high temporal frequencies. The integration extends to a time larger than $t+\Delta t_c$ and a filtering formula, the result of which is centered at time $t + \Delta t_c$, is applied. For example, as shown on Fig. 5, one may extend the integration period to $t + 2\Delta t_c$ and use flat weights over $[t, : t + 2\Delta t_c]$ (constant weights equals to $1/N$ where N is the number of barotropic time steps) to compute the average at time $t + \Delta t_c$.
When filtering is applied, it is relatively easy to interact between grids when no time refinement is done as on Fig. 6 (Barth et al. 2005).
However, when both filtering and time refinement are used (cf. Fig. 7) a new problem arises, due to the fact that some coarse grid points do not have fine grid points located

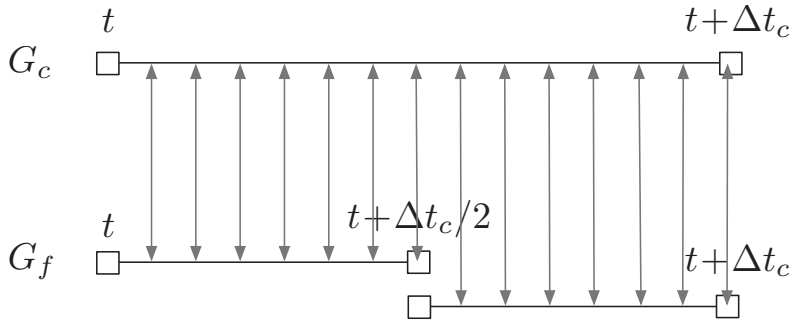


Fig. 4 Time splitting algorithm with a time refinement ratio equal to 2. Intermediate times (due to time refinement) on the fine grid are not shown. G_c is the coarse grid and G_f the fine grid.

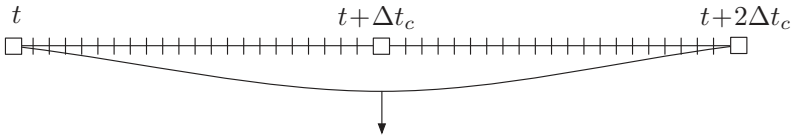


Fig. 5 Time splitting algorithm with filtering using flat weights over $[t, : t + 2\Delta t_c]$. Intermediate instantaneous values are used to compute a filtered value at time $t + \Delta t_c$

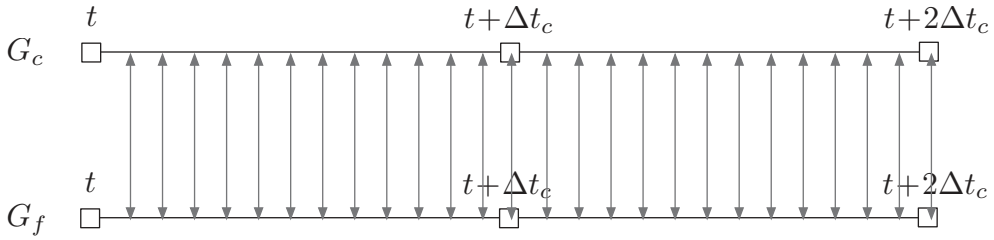


Fig. 6 Time splitting algorithm without time refinement and with filtering

at the same time. For this reason, the coupling between coarse and fine grids cannot be done anymore through the instantaneous barotropic quantities.

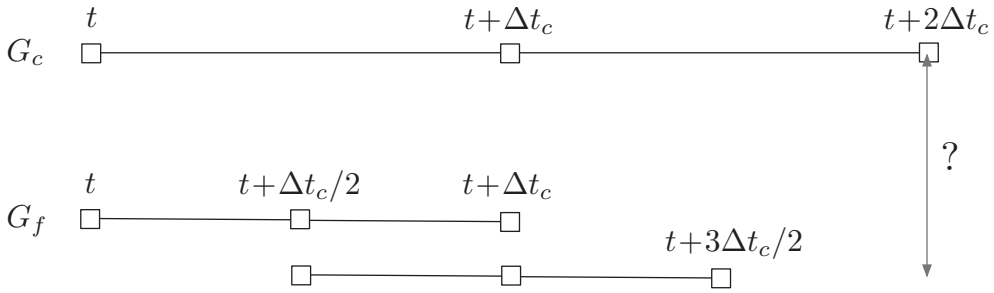


Fig. 7 Time splitting algorithm with time refinement equal to 2 and with filtering

In that case, a simple choice is to perform the coupling at the baroclinic level (Penven 2006), i.e. the exchange occurs only between filtered quantities. However this is like having the system in a one-way mode for the external mode and, as for implicit solvers, errors produced on the coarse grid can then propagate inside the HR grid. One way to perform the coupling at the barotropic level is to exchange information between another set of variables corresponding to intermediate quantities (Debreu et al. 2008). Using intermediate weights, quantities centered at times between t and $t + \Delta t_c$ are

computed so that coupling as without filtering (Fig. 4) is permitted.

Partial conclusion As we have seen, besides the simple algorithm, several additional complexities arise when implicit schemes or time splitting schemes are used. Moreover, it is clear that, in both cases, use of time refinement leads to additional difficulties. Looking at the problem as a truly multilevel system in the case of elliptic systems or doing the coupling at the barotropic level in the case of a time splitting scheme are probably the best choices. Additionally, if more simple techniques are used, then it brings strong constraints on the update operators in order to maintain the coarse grid right hand side in agreement with the fine grid right hand side. Most of the time, these constraints leads to the use of very simple update operators that do not have good numerical properties (see 4.1).

3 Conservation

On a uniform grid, conservation is guaranteed when internal numerical schemes are written in flux form. Concerning two-way embedded models, it is not so common that the resulting system is conservative. Ensuring conservation leads to several computational issues and imposes strong requirements on intergrid transfer operators that can lead to a loss of accuracy. However it is recommended for long term integration. After reviewing the basic requirements for conservation on an embedded grid, two familiar approaches for the preservation of conservation properties are described.

3.1 Definition and discretization

Let us consider a two dimensional domain and q, a solution of the following equation written in conservative form

$$\frac{\partial q}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0$$

where f and g may contain both advective and diffusive fluxes.
Then, assuming that fluxes f and g cancel on the boundaries of Ω, Q_Ω, integral of q over the domain Ω, is constant in time (at the continuous level).

$$Q_\Omega(t) = \int_\Omega q(x,y,t)\,\mathrm{d}x\,\mathrm{d}y \quad \Rightarrow \quad \frac{\mathrm{d}Q_\Omega(t)}{\mathrm{d}t} = \int_{\partial\Omega} f\,\mathrm{d}s + \int_{\partial\Omega} g\,\mathrm{d}s = 0$$

In the nested grid system, the quantity Q_Ω is defined by the summation over the high resolution domain ω and its complement in Ω:

$$Q_\Omega = Q_\omega + Q_{\Omega\setminus\omega}$$

Now let us make the following assumptions for the sake of simplicity:

- As shown on Fig. 8, we consider a two dimensional domain infinite in both x and y directions. The left (resp. right) part of the domain is at coarse (resp. high) resolution.
- The variable $q^n_{i,j}$ is cell centered

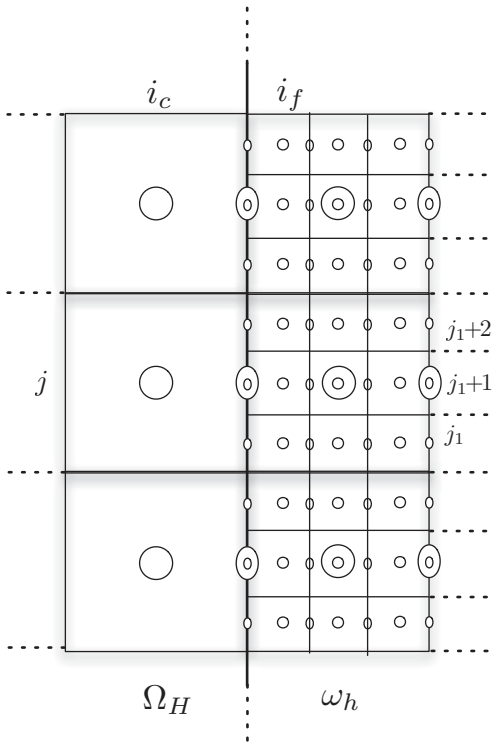


Fig. 8 The CR domain Ω_H on the left and the HR domain ω_h on the right for a C-grid with a mesh refinement factor of 3

– The time stepping scheme is an explicit Euler scheme

$$q_{i,j}^{n+1} = q_{i,j}^{n} - \frac{\Delta t}{\Delta x \Delta y}\left(F_{i,j} - F_{i-1,j}\right) - \frac{\Delta t}{\Delta x \Delta y}\left(G_{i,j} - G_{i,j-1}\right) \tag{4}$$

where $F_{i,j}, G_{i,j}$ are volumetric fluxes: $F_{i,j} = f_{i,j}\Delta y, G_{i,j} = g_{i,j}\Delta x$.
At discrete level, Q_Ω at time $t = t^n$ is given by

$$Q_\Omega^n = \sum_{i \le i_c, j} \Delta x_c \Delta y_c \, q_{i,j}^{c,n} + \sum_{i \ge i_f, j} \Delta x_f \Delta y_f \, q_{i,j}^{f,n}$$

where, as shown on Fig. 8, i_c and i_f denote the first coarse and fine indices close to the interface.
Then after one time step, according to Eq. 4 we obtain

$$\begin{aligned} Q_\Omega^{n+1} &= Q_\Omega^n - \Delta t_c \sum_j F_{i_c,j}^{c,n} + \Delta t_f \sum_{p=0}^{\rho_t - 1} \sum_{j_f = j_1}^{j_1 + \rho - 1} F_{i_f - 1, j_f}^{n + p/\rho_t} \\ &= Q_\Omega^n - \Delta t_c \sum_j \left(F_{i_c,j}^{n} - \frac{1}{\rho_t} \sum_{p=0}^{\rho_t - 1} \sum_{j_f = j_1}^{j_1 + \rho - 1} F_{i_f - 1, j_f}^{n + p/\rho_t} \right) \end{aligned} \tag{5}$$

In general, there will be a misfit between the coarse and fine grid fluxes so that the flux differences of the right hand side do not cancel and conservation is artificially lost: $Q_\Omega^{n+1} \neq Q_\Omega^n$.
Conservation can be achieved if the fine grid fluxes derive from a conservative interpolation (in time and space) of the coarse grid flux so that the following relation holds:

$$\frac{1}{\rho_t} \sum_{p=0}^{\rho_t - 1} \sum_{j_f = j_1 - \rho/2}^{j_1 + \rho/2} F_{i_f - 1, j_f}^{n + p/\rho_t} = F_{i_c,j}^{n} \tag{6}$$

This approach is applied in several ocean models for volume conservation (q being the free surface and F, G the vertically integrated transport in x and y directions). It is indeed sufficient to produce a conservative interpolation of the transport for the global volume to be conserved. A number of classical conservative interpolation schemes can then be used (see 4.2).
Let's consider more generally the computation of a boundary flux on a C-grid as indicated on Fig. 9 and let's suppose than the we are trying to compute the flux of a tracer on the boundary so that:

$$f = uT$$

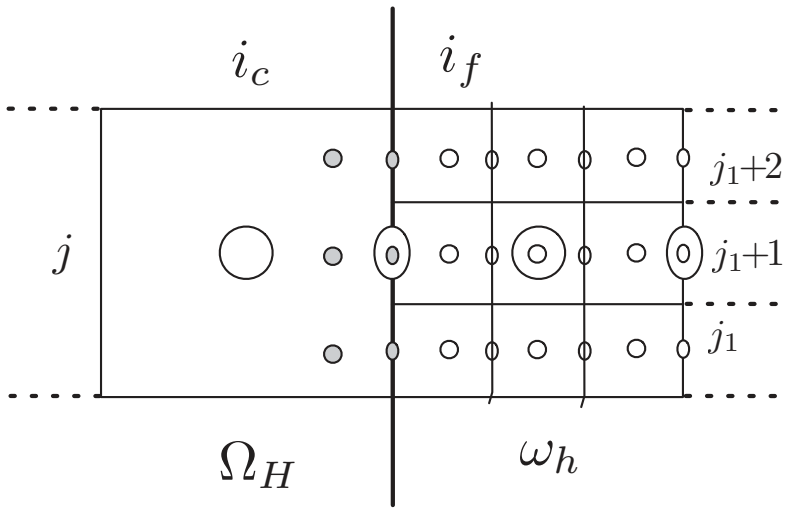


Fig. 9 Computation of a boundary flux on a C-grid

The computation of fine grid flux on the boundary involves the result of the interpolation of the velocity u on the boundary (grey ellipses on Fig. 9). This interpolation has to be conservative in order not to create an artificial loss of conservation. The fine grid flux also involves the computation of the tracer's value on the interface that makes use of both interpolated values (grey circles on Fig. 9) and internal values in the fine grid domain. Thus, we may suppose that the resulting fine grid fluxes are actually more accurate than the coarse grid ones, so that trying to enforce Eq. 6 is not the best choice.

Two other approaches to conservation, that share several similarities, are now introduced: the flux correction algorithm and the Kurihara method.

3.2 Flux correction algorithm

The flux correction algorithm comes from the adaptive mesh refinement community. It follows from the algorithm of Berger and Oliger (1984); Berger and Colella (1989). The idea is to apply a modification of the coarse grid variables that take into account the misfit between coarse and fine grid fluxes.
Starting from Eq. 5, a correction is applied to the coarse grid variable at time $n + 1$

near the boundary as follows :

$$q_{i_c,j}^{n+1,*} = q_{i_c,j}^{n+1} + \frac{\Delta t_c}{\Delta x_c \Delta y_c} \left(F_{i_c,j}^{n} - \frac{1}{\rho_t} \sum_{p=0}^{\rho_t - 1} \sum_{j_f = j - \rho/2}^{j+\rho/2} F_{i_f-1,j_f}^{n+p/\rho_t} \right) \tag{7}$$

Basically it means that the coarse grid variable has been integrated using, on the right interface, fluxes computed by the fine grid :

$$q_{i_c,j}^{n+1,\star} = q_{i_c,j}^{n} - \frac{\Delta t_c}{\Delta x_c \Delta y_c} \left(\frac{1}{\rho_t} \sum_{p=0}^{\rho_t - 1} \sum_{j_f = j - \rho/2}^{j+\rho/2} F_{i_f-1,j_f}^{n+p/\rho_t} - F_{i_c-1,j} \right) \tag{8}$$

The algorithm is easy to implement assuming that it is possible to write the time evolution of q in term of flux divergences as it was indeed the case for the Euler time scheme. The program stores the fine grid fluxes at the boundary and makes a summation in time and in space over the fine grid cells.
With other time stepping schemes, e.g. leap-frog associated to an Asselin filter, this is not achievable (even without the Asselin filter, it is more complex since two arrays of fluxes should be stored, see Herrnstein et al. 2005).

When using this approach, care must be taken concerning potential stability issues. Indeed, if the original fluxes are computed in a centered way, the resulting scheme is always biased and instabilities can occur (Olsson and Petersson 1996; Debreu et al. 2008). One remedy is to compute the interfacial fluxes using information along characteristics of the flow (Part-Enander and Sjogreen 1994, and Blayo and Debreu 2005 for an interpretation of usual open boundary conditions in term of characteristic variables). This is actually the case in the adaptive mesh refinement community where, due to the hyperbolic nature of the problems, the internal numerical schemes already make use of decompositions into characteristic variables.
Moreover, there is an associated loss of accuracy at the boundary. Indeed it can be shown that if the fluxes are computed with a second order accurate approximation, use of Eq. 8 leads to a first order only approximation of the original equation (Debreu et al. 2008).
These two remarks also apply to the Kurihara method that we now describe.

3.3 Kurihara method

The Kurihara method (Kurihara et al. 1979) has been used by several authors to enforce conservation (Kurihara and Bender 1980; Ginis et al. 1998). It is important to note that it applies to a model with all variables cell centered. To understand the method, it is first necessary to explain that the scheme makes use of an intermediate area, composed of two coarse grid cells (grey circles on Fig. 10), where the space resolution is the one of the coarse grid and the time resolution is the one of the fine grid.
In this method, conservation is achieved thanks to two principles :

- The coarse grid computes its fluxes which are then interpolated in time in a conservative way at □ points to force the separation area.

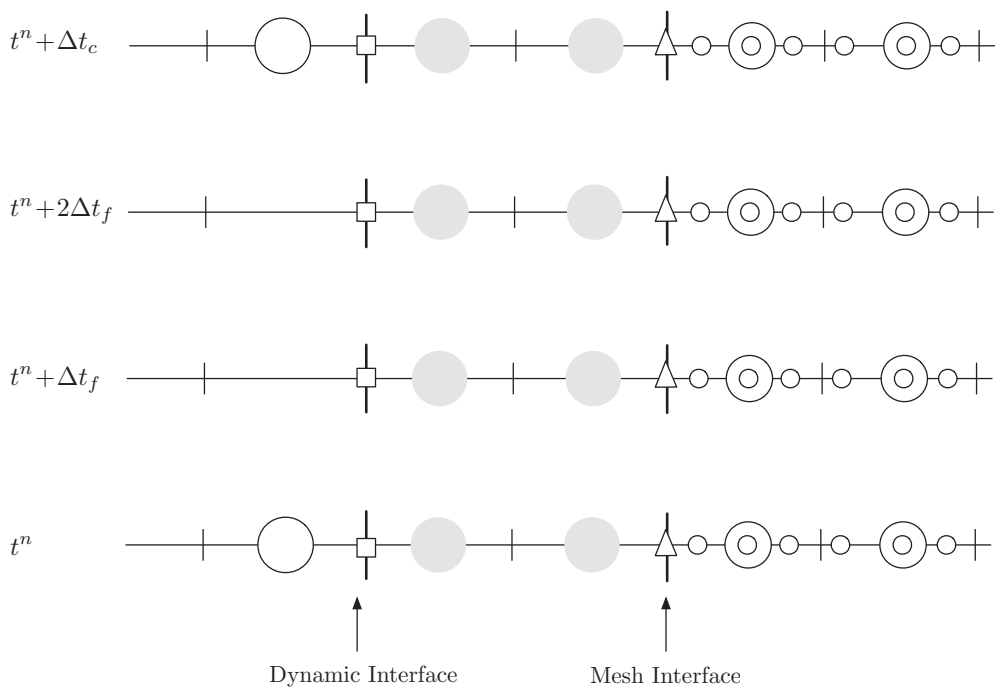


Fig. 10 The three zones of the Kurihara method: on the left the coarse grid, on the right the fine grid. In between, an area composed of two coarse grid cells with the space resolution of the coarse grid and the time resolution of the fine grid.

– A spatial linear interpolation of variables is performed at $\triangle$ points and the computed flux is used both for integrating the coarse grid cell on the left and the fine grid cell on the right. In two dimensions, a summation is applied along the y axis just as for the flux correction approach.

Thus the resulting scheme is conservative. The problems of loss of conservation due to time refinement and to space refinement has been treated separately.
It is clear that if there is no time refinement, then the flux correction approach is equivalent to the Kurihara method if the boundary variables used in the flux computation have also been linearly interpolated.
As already said, the original method is designed for a non staggered grid with all variables located at the cells center. Like for the flux correction approach, the Kurihara method is more difficult to implement for non cell-centered variables (for example in order to preserve conservation of momentum on a C-grid, cf. Sobel 1976).

Partial conclusion We have presented two different approaches for enforcing conservation. Due to the presence of the transition area, the Kurihara method is more difficult to implement than the flux correction approach. Additionally, although possible, it is more difficult to implement such methods for quantities not defined at the centers of the cells. The flux correction approach is easier to implement; however care about potential stability issues must be taken, especially when the original model computes its fluxes using second order centered differences.
When conservation properties are not preserved in a two way simulation, it is important, especially for long term integration, to have an idea of the artificial loss of conservation. That is why the misfit between coarse and fine grid fluxes on the interface should always be diagnosed.

4 Intergrid Transfer operators

For very simple equations (e.g. 1D advection-diffusion), the properties of the two grids methods can be analyzed with simple techniques related to study of numerical schemes on non uniform grids (Harrisson and Elsberry 1972; Berger 1985; Olsson and Petersson 1996). In the linear case, it can even be extended to more complex systems of equations using matrix stability analysis (Heggelund and Bernsten 2002). However in the more general case, the design of intergrid transfer operators must rely on more crude considerations.

4.1 Update schemes

Obviously, the restriction operator has a crucial role in two way nesting algorithms. If we just think in terms of resolved scales, the two following properties should hold :

1. The transfer of information should be maximum for scales well resolved on the coarse grid.
   Otherwise the order of the approximation will be lowered.
2. Small scales should be strongly filtered.
   Otherwise, by aliasing, noise will be produced on the coarse grid.

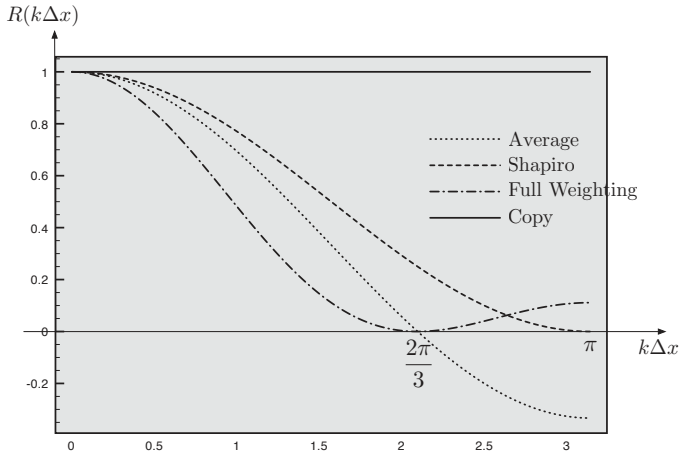
In several applications, authors have made the choice to use the average operator for the feedback step : the coarse grid value in a cell is replaced by the area weighted sum of the fine grid values in the same cell :

$$u^c = \frac{1}{\Delta x^c \Delta y^c} \sum_{i,j} \Delta x^f \, \Delta y^f \, u^f$$

Note that this is often seen as a requirement for conservation but as it has been previously seen, it is not.
In an adaptive mesh refinement context, the restriction is still made with this average formula because the grid can move from one time step to another, so that an average restriction is necessary to maintain globally the conservation before and after the regridding step (periodic modification of the grid hierarchy following a refinement criterion). Note that this was also the main reason for using the average restriction operator in the Kurihara paper (Kurihara et al. 1979), where the methodology is applied to moving mesh methods.
Other familiar restriction schemes are written on Fig. 11 along with their corresponding transfer functions.
Recently in Debreu et al. (2008), it is shown that use of higher order restriction operators, using a larger stencil (full weighted schemes), can lead to strong improvements. Even if the full weighted operator damps the well resolved scales a little more, it also produces a strong damping of the small scales which is required to inhibit aliasing and noise on the coarse grid.

4.2 Interpolation

Numerous interpolation operators have been used in two way nesting methods. See Koch and McQueen (1987); Zhang et al. (1986) for the details in several models. The



Copy	$u^c = u^f$
Average	$u^c = \frac{1}{3}\left(u^f_{j-1} + u^f_j + u^f_{j+1}\right)$
Shapiro	$u^c = \frac{1}{4}\left(u^f_{j-1} + 2u^f_j + u^f_{j+1}\right)$
Full Weighting	$u^c = \frac{1}{9}\left(u^f_{j-2} + 2u^f_{j-1} + 3u^f_j + 2u^f_{j+1} + u^f_{j+2}\right)$

Fig. 11 Transfer functions for usual one dimensional restriction operators with a mesh refinement factor $\rho = 3$ for typical filters (k is the wavenumber). The corresponding formula are also given.

basic idea is that the order of the interpolation order should be high enough not to lower the global order of approximation and, as usual when using high order interpolation schemes, care must be taken about the potential oscillations produced by the schemes (this problem is well known, for example, when using Lagrange interpolation operators). As previously seen, it is also advantageous to have conservative interpolations (in particular for the interpolation of normal velocities on the boundaries). To fulfill these two conditions several methods can be used: parabolic conservative interpolation or approximation Clark and Farley (1984), parabolic interpolation with minimization of second order derivative Barth et al. (2005), PPM (Piecewise Parabolic Method), WENO (Weighted Essentially Non Oscillatory), advection equivalent interpolation schemes Alapaty et al. (1998) ...

Several authors also mention the fact that interpolation and restriction operators must be devised in agreement one with the other. This is mainly justified when implicit solvers are used. The main reason is that, after feedback, the right hand side of the elliptic equation, computed on the coarse grid, must be in agreement with the one of the fine grid (Clark and Farley 1984). In that case, assuming a conservative interpolation has been used, the average restriction operator is the most natural. Let's take the example of the free surface equation written under the following form (Eq. 9):

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (U, V) = 0 \tag{9}$$

where η is the free surface and (U, V) the barotropic transports in x and y directions. If the transport U has been conservatively interpolated on the fine grid boundary, then a simple update (average in one direction and simple copy in the other direction as

shown on Fig. 12) of the transports leads to a barotropic divergence in the coarse grid cell equal to the average of the barotropic divergence on the fine grid cell:

$$\nabla \cdot (U^c, V^c) = \frac{1}{9} \sum_{i,j} \nabla \cdot (U_i^f, V_j^f) \tag{10}$$

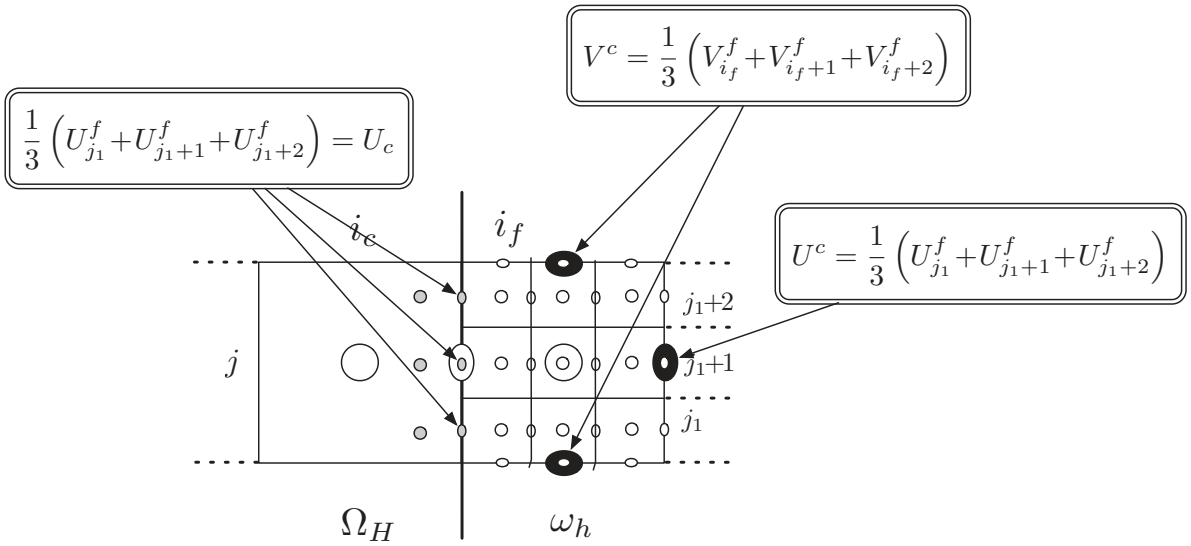


Fig. 12 Update and interpolation schemes for the preservation of the barotropic divergence on a C-grid with a mesh refinement factor of 3. The transport (U) are conservatively interpolated on the boundary (grey ellipses) and then simple update operators are used at coarse grid points (black ellipses).

However, as we already saw (paragraph 2.2.1), the constraint given by Eq. 10, which actually corresponds to an equality between tendencies of the free surface equation instead of instantaneous values, can be removed if either the right hand side of the coarse grid is directly updated by the right hand side of the fine grid or if a truly multilevel solver is used. Removing this constraint allows the use of more scale selective update operators than those used here.

4.3 Inconsistency at boundary

We now look at a particular problem that also induces differences between implementation of intergrid transfer operators.
Let's rewrite the explicit algorithm given in paragraph 2.1

1. $u_c^{n+1} = \mathcal{L}_c(u_c^n)$
2. `For` $m = 1 \dots \rho_t$ `do`
   $$u_f^{n+\frac{m}{\rho_t}} = \mathcal{L}_f\left(u_f^{n+\frac{(m-1)}{\rho_t}}\right)$$
   $$u_f^{n+\frac{m}{\rho_t}}{}_{|\gamma_h} = P(u_c^n, u_c^{n+1})$$
3. $\mathbf{u_c^{n+1}}_{|\omega_H} = R(u_f^{n+1}(\mathbf{u_c^{n+1}}))$

where we insist on the fact that u_f^{n+1} is indeed dependent on u_c^{n+1} through the boundary interpolation so that an inconsistency (sometimes called overspecification) is inherent to the third step. Note that this does not apply to variables entirely lying on the interface, like normal velocities on a $C-$ grid, because the coarse grid variables used in the interpolation (along the interface) are not updated in the feedback step. On a $C-$grid, this inconsistency occurs for tracers and tangential velocities. One possible way to remove this inconsistency is to modify the interpolation step 2 by doing intermediate updates so that step 2 is modified as follows

$$u_f^{n+\frac{m}{\rho_t}}{}_{|\gamma_h} = P(u_c^n, u_c^{n+1}) \quad \rightarrow \quad u_f^{n+\frac{m}{\rho_t}}{}_{|\gamma_h} = P(u_c^n, \left\{ \begin{array}{ll} u_c^{n+1} & \text{in } \Omega_H \backslash \omega_H \\ R(u_f^{n+\frac{m}{\rho_t}}{}_{|\gamma_h}) & \text{in } \omega_H \end{array} \right\})$$

eventually using a fixed point iteration if the operators P and/or R are non linear.
To our knowledge, this method has not been explored. In the literature, several other methods have been developed to try to remove this contradiction:

– Interpolation operators
It is possible to write the interpolations so that no coarse grid point, that will be modified during the feedback step (i.e. inside the HR grid domain) are used in the interpolation process. In order to do this, a tangential interpolation is first applied, followed by a normal interpolation. The idea is also to use the HR solution as soon as available thus lowering interpolation errors (both in space and time).
Note that here too, care must be taken about stability issues. As an example, using a high order interpolation, which involves several fine grid points inside the fine grid domain, biases the interpolation schemes and can produce unstable solutions if the flow is actually entering the fine grid. So that in practice the normal interpolation should be computed using the direction of the flow. Again this is especially true when the model computes its fluxes using second order centered differences.

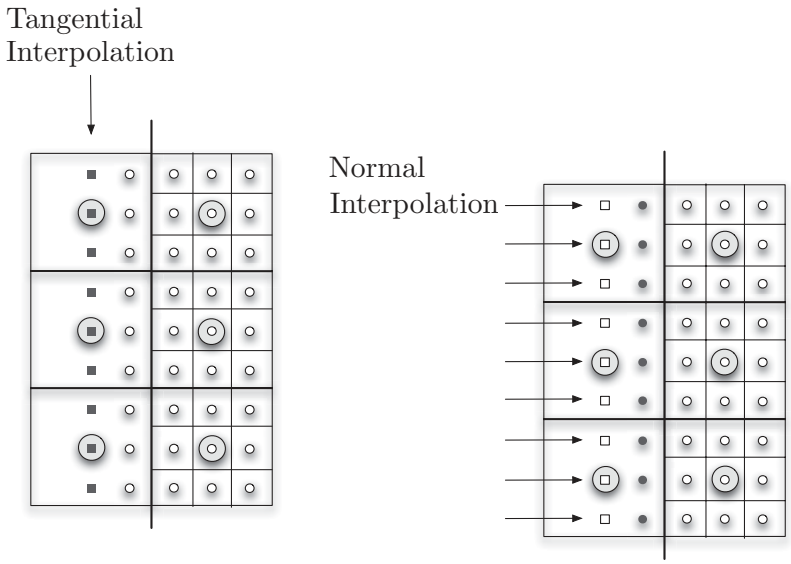


Fig. 13 Tangent interpolation followed by a normal interpolation. A set of fictitious points (squared boxes) are first obtained by interpolation along the tangential direction, this is followed by a normal interpolation that, in addition to these points, uses fine grid points inside the fine grid domain. Note that the coarse grid points inside the fine grid domain are not used in the interpolation process.

– Update operators - Separation of dynamic and feedback interface
Another remedy to remove this inconsistency is to modify the feedback step. Several authors have proposed to separate the feedback interface from the dynamic interface (sometimes also called input interface) where the boundary values are interpolated (Phillips and Shukla 1973; Zhang et al. 1986; Oey and Chen 1992). Similarly, the Kurihara method can also be seen as introducing a separation between forcing and feedback interfaces. Figure 14 shows a separation of the dynamic and feedback interface by one coarse grid cell. This separation can also be composed of two coarse grid cells. In these cases, the contradiction has been removed since the coarse grid points used in the interpolations are not updated.

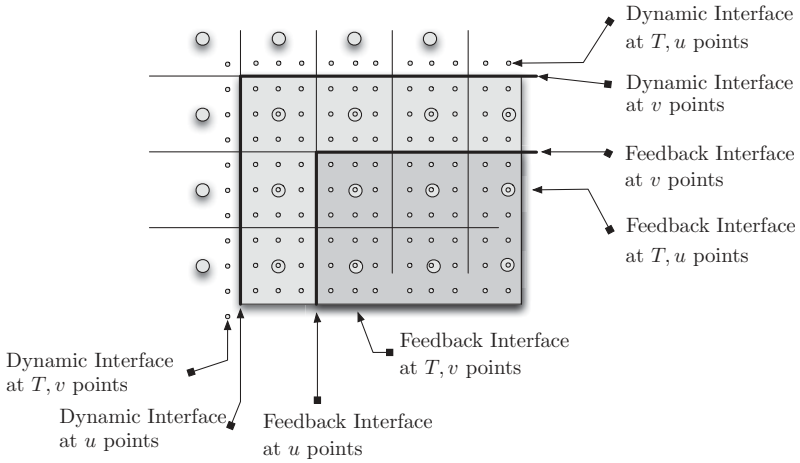


Fig. 14 Separation of dynamic and feedback interface on a $C-$ grid for a mesh refinement factor of 3. Without a separation of these interfaces, the tracers and tangential velocities would also be updated in the light grey area.

Another reason for using a mesh separation between dynamic and feedback interfaces is that if noise is produced, it will be larger near the dynamic interface so that it is safer not to use the fine grid values near the dynamic interfaces.
However, as mentioned by Spall and Holland (1991), setting the feedback interface closer to the dynamic interface maximizes the information transferred to the coarse grid. In addition since there is no feedback in this area, the consistency between coarse and fine solutions will be less than with feedback and this inconsistency has to be lowered by use of noise control techniques that will indeed recombine and smooth the HR and CR solutions in the area between dynamic and feedback interfaces.

Partial conclusion From the authors' experience, update operators have a greater impact on the quality of the solution than interpolation operators (assuming that these last ones are properly built, e.g. conservative when they have to be). The usual update operators (average, Shapiro filter ...) do not have sufficiently good numerical properties to lead to stable simulations. In particular, use of update operators that produce a stronger damping of small scales is required.
The technique of separation of dynamic and feedback interfaces can be used associated with noise control techniques to make robust two way embedding simulations. However errors introduced by these methods may be found to be very large when quantitatively evaluated.

5 Noise Control

Two principles can motivate the use of noise control techniques :

- Maintain a strong consistency between HR and CR solutions in the area where solutions interact (i.e. near the common interface).
- Prevent waves reflection: waves unresolved on the coarse grid reflect inside the fine grid domain and have to be filtered in some way.

It is generally accepted that one criterion for choosing the noise control technique is that the associated modifications should cancel if the refinement ratio is equal to 1, so that if the simulation is made with a mesh refinement factor of 1, the result is identical to the result computed with a uniform grid.
Of course, use of noise control techniques modify the original differential operator near the interface and thus lower the order of approximation.
Two-way embedding models can make use of noise control techniques that are traditionally used in open boundary and/or one way embedded simulations. We briefly recall here some of them:

- Use of a time stepping algorithm or other internal numerical schemes that damps small wavelengths. This is the crudest form of noise control technique: the original model solution is sufficiently damped so that small scales are strongly removed.
- Relaxation methods
  - Blending (Flow Relaxation Scheme)

    $$q_f^\star = (1 - \mu_{x,\partial\omega}) q_f + \mu_{x,\partial\omega}\ P q_c$$

    $\mu_{x,\partial\omega}$ is a coefficient varying from 1 on the interface $\partial\omega$ to 0 away to the interface (Davies 1976). This is used in Oey and Chen (1992) in the separation area between feedback and dynamic interfaces.
  - Nudging

    $$\frac{\partial q_f}{\partial t} = \ldots - \frac{(q_f - P q_c)}{T_d}$$

    The discrete form of nudging is equivalent to the flow relaxation scheme (Martinsen and Engedahl 1987).
- Increase of dissipation coefficient - Sponge Layer
  When using a sponge layer it is advantageous to apply it to the difference between external (coarse resolution) and internal (high resolution) fields

  $$\frac{\partial q_f}{\partial t} = \ldots + (-1)^{n+1} (\Delta)^n \left[\mu_{x,\partial\omega} (q_f - P q_c) \right] \tag{11}$$

  Here the objective is clearly to damp small scales as defined by the restriction operator. Since the coarse grid variable q_c has been updated ($q_c = \mathcal{R} q_f$), the diffusion term mostly acts on the scales lying in the kernel of the restriction operator ($\mathcal{R} q_f = 0$). Note that the preceding relaxation methods can be put under this category just by choosing $n = 0$ in Eq. 11 so that they produce a less selective damping than for $n \geq 1$.
  The advantage of this formulation is also that the diffusion term naturally cancels if there is no refinement. Using differences between coarse and fine grid variables can also be at the basis of the derivation of open boundary schemes suitable for embedding methods (e.g. Perkins et al. 1997, Oddo and Pinardi 2008).

Additionally, in a flux correction approach of conservation, it is easy to maintain conservation just by adding the diffusive flux of Eq. 11.

Moreover, due to the inability to build ideal restriction operators, it may also be useful to smooth the coarse grid solution inside the fine grid domain. The filtering is only applied to the coarse grid points inside the fine grid domain in order not to violate conservation principles. As an example, the smoothing-desmoothing algorithm of Bender et al. (1993) can be applied on the first coarse grid points near the interfaces.

Partial conclusion Two way grid embedding algorithms should be first evaluated without using noise control techniques in order not to hide fundamental problems of the algorithm. Because the primary requirement is that the associated modifications of the original model should cancel if the refinement factor is equal to 1, most of the methods are formulated in term of differences between coarse and fine fields as in Eq. 11. Choosing $n = 1$ in Eq. 11 (laplacian diffusion) is a good compromise between a low computational cost and a selective damping of the difference. For $n \geq 1$, the correction can be easily written in term of flux divergences thus allowing the conservation properties to be maintained (e.g. using the flux correction approach).

6 Conclusion

This paper reviews existing two way embedding techniques. The emphasis has been put on what are the sources of potential approximations and errors. In particular, we have shown how the use of implicit solver and/or time splitting algorithms and the use of refinement in time can introduce several problems in comparison with an idealized model. Conservation issues can be tackled in a simple way, using the flux correction approach (at least for quantities based on cell centered variables: mass, tracers ...). There is also a large choice of intergrid transfer operators, the choice between one of them is a trade-off between conservation, accuracy, robustness and coding complexities. Some guidelines have been given as partial conclusions of each section.
Several issues related to grid embedding have not been covered. Computational implementation of embedding can be done by hand or by using dedicated software (e.g. AGRIF Debreu et al. 2008, RSL Michalakes 1998). In the context of operational oceanography and production of high resolution forecasts at particular areas of interest, high resolution grid initialization schemes which are in agreement with the two way interactions have to be derived. As most of the operational systems are now using data assimilation methods, one possible way to look at the problem is to introduce the high resolution grid initialisation in the assimilation process and this is an active research subject.

Acknowledgements This work is a contribution to the MERSEA project. Partial support from the European Commission under contract SIP3-CT-2003-502885 is gratefully acknowledged. We would like to thank two anonymous reviewers for their comments and careful reading of the manuscript.

References

Alapaty K, Mathur R, Odman T (1998) Intercomparaison of Spatial Interpolation Schemes for Use in Nested Grid Models. Mon Weather Rev 126(1):243-249

Barth A, Alvera-Azcarate A, Rixen M, Beckers J-M (2005) Two-way nested model of mesoscale circulation features in the Ligurian Sea. Prog Oceanogr 66:171-189.

Bender MA, Ross RJ, Tuleya RE, Kurihara Y (1993) Improvements in tropical cyclone track and intensity forecasts using the GFDL initialization system. Mon Weather Rev 121:2046-2061

Berger MJ (1985) Stability of Interfaces with Mesh Refinement. Math Comput 45:301-318

Berger MJ, Oliger J (1984) Adaptive mesh refinement for hyperbolic partial differential equations. J Comput Phys 53(3):484-512

Berger MJ, Colella P (1989) Local adaptive mesh refinement for shock hydrodynamics. J Comput Phys 82(1):64-84

Blayo E, Debreu L, (2005) Revisiting open boundary conditions from the point of view of characteristic variables. Ocean Model 9:231-252

Blumberg AF, Mellor GL (1987) A description of a three-dimensional coastal ocean circulation model. Three-Dimensional Coastal Ocean Models, Vol 4., N. Heaps, Ed., American Geophysical Union, 208pp.

Clark TL, Farley D (1984) Severe downslope windstorm calculations in two and three spatial dimensions using anelastic interactive grid nesting: A possible mechanism for gustiness. J Atmos Sci 41:329-350

Davies, HC, (1976) A lateral boundary formulation for multi-level prediction models. Q J R Meteorol Soc 102:405-418

Debreu L, Vouland C, Blayo E (2008) AGRIF: Adaptive Grid Refinement In Fortran. Comput Geosci-Uk 34(1):8-13

Debreu L, Marchesiello P, Penven P (2008) Two-Way embedding algorithms for a split-explicit free surface ocean model. In preparation for Ocean Model.

Dukowicz JK, Smith RD, (1994) Implicit free-surface method for the Bryan-Cox-Semtner ocean model. J Geophys Res 99:7991:8014

Fox AD, Maskell SJ, (1995) Two-way interactive nesting of primitive equation ocean models with topography. J Phys Oceanogr 25:2977-2996

Ginis I, Richardson RM, Rothstein LM (1998) Design of a multiply nested primitive equation ocean model. Mon Weather Rev 126:1054-1079

Harrison EJ, Elsberry RL (1972) A method for incorporating nested finite grids in the solution of systems of geophysical equations. J Atmos Sci 29:1235-1245

Heggelund Y, Bernsten J (2002) A method for analysing nesting techniques for the linearized shallow water equations. Int J Numer Meth Fl 38:163-185

Herrnstein A, Wickett M, Rodrigue G (2005) Structured adaptive mesh refinement using leapfrog time integration on a staggered grid for ocean models. Ocean Model 9(3):283-304

Jones RW (1977) A nested grid for a three-dimensional model of a tropical cyclone. J Atmos Sci 34:1528-1553

Killworth PD, Stainforth D, Webb DJ, Paterson SM (1991) The development of a free-surface Bryan-Cox-Semtner ocean model. J Phys Oceanogr 21:1333-1348

Koch SE, McQueen JT (1987) A survey of Nested Grid Techniques and Their Potential for Use Within the MASS Weather Prediction Model. NASA Technical Memorandum 87808.

Kurihara Y, Tripoli GJ, Bender MA (1979) Design of a movable nested-mesh primitive equation model. Mon Weather Rev 107:239-249

Kurihara Y, Bender MA (1980) Use of a movable nested-mesh model for tracking a small vortex. Mon Weather Rev 108:1792:1809

Laugier M, Angot P,Mortier L (1996) Nested grid methods for an ocean model: a comparative study. Int J Numer Meth Fl 23(11):1163-1195

Martin DF, Cartwright KL (1996) Solving Poisson's equation using adaptive mesh refinement. Technical Report UCB/ERL, M96/66 UC, Berkely.

Martinsen EA, Engedahl HE (1987) Implementation and testing of a lateral boundary scheme as an open boundary condition in a barotropic ocean model. Coast Eng 11:603-627

Michalakes J (1998) Runtime System Library (RSL) for parallelization of gridded weather models with nesting. http://www-unix.mcs.anl.gov/~michalak/rsl

Oddo P, Pinardi N (2008) Lateral open boundary conditions for nested limited area models: A scale selective approach. Ocean Model 20:134-156

Oey LY, Chen P (1992) A nested-grid ocean model: with application to the simulation of meanders and eddies in the Norwegian Coastal Current. J Geophys Res 97(20):063-086

Olsson F, Petersson NA (1996) Stability of interpolation on overlapping grids. Comput Fluids 25(6):583-605

Part-Enander E, Sjogreen B (1994) Conservative and non-conservative interpolation between overlapping grids for finite volume solutions of hyperbolic problems. Comput Fluids 23(3):551-574

Penven P, Debreu L, Marchesiello P, McWilliams JC (2006) Evaluation and application of the ROMS 1-way embedding procedure to the central california upwelling system. Ocean Model 12:157-187

Perkins AL, Smedstad LF, Blake DW, Heburn GW, Wallcraft AJ (1997) A new nested boundary condition for a primitive-equation ocean model. J Geophys Res 102(C2):3483-3500.

Phillips NA, Shukla J (1973) On the strategy of combining coarse and fine grids meshes in numerical weather prediction. J Appl Meteorol 12:763-770

Sheng J, Greatbatch RJ, Zhai X, Tang L (2005) A new two-way nesting technique for ocean modeling based on the smoothed semi-prognostic method. Ocean Dyn 55:162-177

Sobel JP, (1976) Nested grids in numerical weather prediction and an application to a mesoscale streak. PhD. thesis, Pennsylvania State University, 135pp.

Spall MA, Holland WR (1991) A nested primitive equation model for oceanic applications. J Phys Oceanogr 21:205-220

Zhang DL, Chang HR, Seaman NL, Warner TT, Fritsch JM (1986) A Two-Way Interactive Nesting Procedure with Variable Terrain Resolution. Mon Weather Rev 114:1330-1339

Zhuo L, Qingcun Z (1995) The Application of Time-Dependent Adaptive Mesh Model in Typhoon Track Prediction. Chinese J Atmos Sci 19:303-310.