

HENRY

Hydraulic Engineering Repository

Ein Service der Bundesanstalt für Wasserbau

Article, Author's Postprint

Sehili, Aissa; Lang, Günther; Lippert, Christoph

High Resolution Subgrid Models: Background, Grid generation and Implementation

Ocean Dynamics

Verfügbar unter/Available at: <https://hdl.handle.net/20.500.11970/104573>

Vorgeschlagene Zitierweise/Suggested citation:

Sehili, Aissa; Lang, Günther; Lippert, Christoph (2013): High Resolution Subgrid Models: Background, Grid generation and Implementation. In: Ocean Dynamics Issue 4. S. 519-535.

Standardnutzungsbedingungen/Terms of Use:

Die Dokumente in HENRY stehen unter der Creative Commons Lizenz CC BY 4.0, sofern keine abweichenden Nutzungsbedingungen getroffen wurden. Damit ist sowohl die kommerzielle Nutzung als auch das Teilen, die Weiterbearbeitung und Speicherung erlaubt. Das Verwenden und das Bearbeiten stehen unter der Bedingung der Namensnennung. Im Einzelfall kann eine restriktivere Lizenz gelten; dann gelten abweichend von den obigen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Documents in HENRY are made available under the Creative Commons License CC BY 4.0, if no other license is applicable. Under CC BY 4.0 commercial use and sharing, remixing, transforming, and building upon the material of the work is permitted. In some cases a different, more restrictive license may apply; if applicable the terms of the restrictive license will be binding.



Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

Erstveröffentlichung in *Ocean Dynamics* 64 (2014), S. 519-535.

Für eine korrekte Zitierbarkeit ist die Seitennummerierung der Originalveröffentlichung für jede Seite kenntlich gemacht.

S. 519

High-resolution subgrid models: background, grid generation, and implementation

Aissa Sehili^a · Günther Lang^b · Christoph Lippert^c

^a Federal Waterways Engineering and Research Institute, Wedeler Landstr. 157, 22559 Hamburg, Germany

e-mail: aissa.sehili@baw.de

^b e-mail: guenther.lang@baw.de

^c Smile consult GmbH, Vahrenwalder Str. 4, 30165 Hannover, Germany

e-mail: lippert@smileconsult.de

Responsible Editor: Pierre Lermusiaux

Received: 11 June 2013 / Accepted: 18 January 2014 / Published online: 11 March 2014

The basic idea of subgrid models is the use of available high-resolution bathymetric data at subgrid level in computations that are performed on relatively coarse grids allowing large time steps. For that purpose, an algorithm that correctly represents the precise mass balance in regions where wetting and drying occur was derived by Casulli (Int J Numer Method Fluids 60:391–408, 2009) and Casulli and Stelling (Int J Numer Method Fluids 67:441–449, 2010). Computational grid cells are permitted to be wet, partially wet, or dry, and no drying threshold is needed. Based on the subgrid technique, practical applications involving various scenarios were implemented including an operational forecast model for water level, salinity, and temperature of the Elbe Estuary in Germany. The grid generation procedure allows a detailed boundary fitting at subgrid level. The computational grid is made of flow-aligned quadrilaterals including few triangles where necessary. User-defined grid subdivision at subgrid level allows a correct representation of the volume up to measurement accuracy. Bottom friction requires a particular treatment. Based on the conveyance approach, an appropriate empirical correction was worked out. The aforementioned features make the subgrid technique very efficient, robust, and accurate. Comparison of predicted water levels with the comparatively highly resolved classical unstructured grid model shows very good agreement. The speedup in computational performance due to the use of the subgrid technique is about a factor of 20. A typical daily forecast can be carried out in less than 10 min on a standard PC-like hardware. The subgrid technique is therefore a promising framework to perform accurate temporal and spatial large-scale simulations of coastal and estuarine flow and transport processes at low computational cost.

Keywords: Subgrid modeling, Wetting and drying, Semi-implicit schemes, Operational forecast models

1 Introduction

The simulation of coastal flows involving wetting and drying of large tidal areas implies two major difficulties: (i) the need for stable, accurate, and efficient algorithms and (ii) the use of model grids that are able to reproduce the bathymetric details and boundaries up to an acceptable level. A general review of popular wetting and drying algorithms is given by Balzano (1998) and Van't Hof and Vollebregt (2005). Over the last two decades, a family of numerical models for solving 2D and 3D shallow water equations based on the efficient and robust semi-implicit finite difference method has been proposed by Casulli and his coauthors (Casulli 1990; Casulli and Cheng 1992; Casulli and Cattani 1994; Casulli and Walters 2000). The models range from the 2D structured linear Tidal Residual and Intertidal Mudflat (TRIM) model up to the 3D unstructured piecewise linear UnTRIM model. The need to capture bathymetric details in complex regions, with a better boundary fit and to reduce the grid resolution in large and open regions, like tidal

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 520

flats, motivated the transition from structured to unstructured orthogonal grids. What is new, on the other hand, is that modern remote sensing technologies can deliver very detailed land surface height data that should be considered for more accurate simulations. In that case, and even if some compromise is made with regard to grid resolution of an unstructured grid, simulations still will require large grids which can be computationally very demanding.

Recently, new techniques have been proposed for flood simulation such as the raster-based models working with high-resolution topographic data (Bates and De Roo 2000; Horritt and Bates 2001; Yu and Lane 2006a), or models that make use of subgrid-scale details to a limited extent (Bates 2000; Yu and Lane 2006b). The subgrid technique, first published by Casulli (2009), is based on the idea of making use of the available detailed subgrid bathymetric information while performing computations on relatively coarse grids permitting large time steps. Consequently, accuracy and efficiency are drastically enhanced. Compared to the classical linear method, where the underlying bathymetry is solely discretized by the computational grid, the resulting system is mildly nonlinear requiring only few extra Newton iterations to be solved. The algorithm guarantees rigorous mass conservation and nonnegative water depths for any time step size. The generalization of the semi-implicit algorithm for 3D flows (Casulli 2009) to intrinsically account for detailed bathymetric subgrid data was demonstrated by Casulli and Stelling (2010). A special treatment, proposed in the present work, is applied to bottom friction for friction dominated flows. Based on the subgrid technique, flood simulations with subgrid digital elevation models were performed (Stelling 2012). The aforementioned features motivated the decision to use the UnTRIM² subgrid system for the simulation of the Elbe Estuary system.

The subgrid technique is accompanied by new challenges associated with issues of grid (subgrid) generation. For that, new techniques have been developed. The present paper aims to give some insights in the subgrid technique and to shed some light on issues like grid generation and numerical efficiency. One implementation of the technique at the Federal Waterways Engineering and Research Institute (BAW) in an operational forecast model for the Elbe Estuary in Germany will be presented.

2 The subgrid technique

Unstructured orthogonal grid The shallow water equations are defined on a time-dependent domain $\Omega(t)$ defined as $\Omega(t) = \{(x, y) : H(x, y, t) > 0\}$. The whole domain Ω is represented by a set of N_p nonoverlapping convex polygons Ω_i , $i = 1, 2, \dots, N_p$ involving N_s sides (see Fig. 1). Within each polygon, a center has to be identified in such a way that the segment joining the centers of two adjacent polygons and the side shared by the two polygons have a nonempty intersection and are orthogonal to each other. Each polygon may have an arbitrary number of sides. In the 3D case, the domain is discretized vertically using a fixed number of horizontal surfaces (z-planes) which can be nonuniformly distributed. Polygons transform to prisms having the thickness of the corresponding z-plane. In a staggered C grid, the water surface elevation is located at the center of the polygon. The velocity component normal to each face (edge) is located at the face (edge) center. We refer to Casulli and Walters (2000) for a comprehensive description

2.1 Basic principles of subgrid resolution

In order to aid later discussion, we recall here the basic principles of subgrid resolution as introduced by Casulli (2009). During wetting and drying, the boundary of the computational domain moves and is a priori unknown. For any specified bathymetry $h(x, y)$ and in order to exactly represent the free-surface area, a porosity function $p(x, y, z)$ is defined by

$$p(x, y, z) = \begin{cases} 1 & \text{if } h(x, y) + z > 0 \\ 0 & \text{elsewhere} \end{cases} \quad (1)$$

with $(x, y) \in \Omega$ and $-\infty < z < +\infty$. Within each polygon i having an area P_i , the horizontal integral of the porosity evaluated at the free surface $z = \eta_i^n$ at time step n is given by

$$p_i(\eta_i^n) = \int_{\Omega_i} p(x, y, \eta_i^n) dx dy \quad (2)$$

Equation 1 implies that $p_i(\eta_i^n)$ is nonnegative, nondecreasing, and bounded. When $p_i(\eta_i^n) = 0$, the i th polygon is dry; when $p_i(\eta_i^n) = P_i$, it is wet; and when $0 < p_i(\eta_i^n) < P_i$, the i th polygon is partially wet. For each location within the i th polygon, the water depth is given by

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

$$H(x, y, \eta_i^n) = \int_{-\infty}^{\eta_i^n} p(x, y, z) dz = \max[0, h(x, y) + \eta_i^n] \quad (3)$$

with $H(x, y, \eta_i^n) \geq 0$. Consequently, a wet region Ω_i within the i th polygon satisfies

$$\Omega_i^n = \{(x, y) \in \Omega_i : H(x, y, \eta_i^n) > 0\} \quad (4)$$

The water volume V_i within the i th polygon can be expressed either as a horizontal integral of the water depth or as a vertical integral of the surface wet area p_i :

$$V_i(\eta_i^n) = \int_{-\infty}^{\eta_i^n} p_i(z) dz = \int_{\Omega_i} H(x, y, \eta_i^n) dx dy \quad (5)$$

Hence, $V_i(\eta_i^n) \geq 0$ since $p_i(z)$ is nonnegative and nondecreasing. If $V_i(\eta_i^n) > 0$, then necessarily $p_i(\eta_i^n) > 0$. This equation induces the nonlinearity of the subgrid method.

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 521

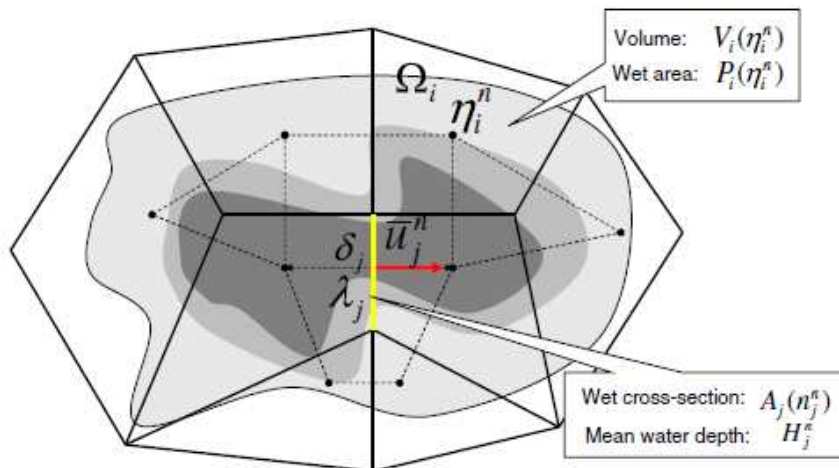


Fig. 1 Unstructured orthogonal grid with subgrid information

The water volume is no more a linear function of the water depth given that the latter varies with the prescribed subgrid depth $h(x, y)$ inside a computational polygon as formulated in Eq. 3. This is particularly true when the polygon is partially wet. In dry polygons, the volume is zero; in partially wet polygons (transition cells), the volume is a nonlinear function of water depth. In fully wet poly-

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

gons, the volume becomes a linear function of water depth (see linear versus nonlinear paragraph in the following). The nonnegative cellaveraged water depth can be defined as $H_i^n = V_i(\eta_i^n)/P_i$. Moreover, for a specified constant water level η_j^n along the j th edge, the wet cross-section area is given by

$$A_j(\eta_j^n) = \int_{s_j^1}^{s_j^2} H(x(s), y(s), \eta_j^n) ds \quad (6)$$

where $x(s)$ and $y(s)$ are the parametric coordinates of a point on the j th edge connecting the two vertices identified by the parameters $s = s_j^1$ and $s = s_j^2$. The nonnegative edgeaveraged water depth can be defined as $H_j^n = A_j(\eta_j^n)/\lambda_j^n$. The wet edge length λ_j^n is no longer constant and depends on the elevation. The wet edge length λ_j^n is no longer constant and depends on the elevation η_j^n .

The previous basic principles of subgrid specification are used to solve the momentum and mass conservation equations by means of the efficient semi-implicit scheme whose stability is independent from the free-surface wave speed, vertical viscosity, and bottom friction (Casulli 1990). The resulting finite volume formulation leads to a mildly nonlinear system of equations for the discrete free-surface elevation. Along with a converging Newton-type iteration, this large system is solved every time step with an efficient preconditioned conjugate gradient solver whose convergence can be guaranteed under physically compatible assumptions.

Linear versus nonlinear approach According to Eq. 5, the major difference between the subgrid approach and the classical method like the one used in linear or piecewise linear semi-implicit methods arises in the presence of wetting and drying. As long as the free surface does not intersect with the bathymetry, the water volume remains linearly dependent on water depth, and the subgrid specification has no influence on the resulting flow which is determined only by the edge-averaged bathymetry h_j . Indeed, from Eqs. 3 and 5, one gets

$$\begin{aligned} V_i(\eta_i^n) &= \int_{\Omega_i} \max[0, h(x, y) + \eta_i^n] dx dy \\ &\geq \int_{\Omega_i} [h(x, y) + \eta_i^n] dx dy = P_i(h_i + \eta_i^n). \end{aligned} \quad (7)$$

The inequality $V_i(\eta_i^n) \geq P_i(h_i + \eta_i^n)$ becomes strict in the presence of wetting and drying. That means that water volume in the case of the subgrid nonlinear formulation is always larger than or equal to the one delivered by the classical linear or piecewise linear approach. The piecewise linear formulation, which is the precursor of the subgrid technique, can be seen as a particular case of the nonlinear subgrid method if the grid is prescribed with only one subpolygon and only one subedge. Unlike the linear formulation, which allows unphysical negative water depths, the nonlinear subgrid formulation always achieves correct mass balance in wet cells, as well as in partially wet cells, independently of the specified bathymetry. Nonnegative water volumes and nonnegative water depths are always assured everywhere. Additionally, the specification of highly resolved bathymetry at

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

subgrid level allows an accurate mass balance during wetting and drying since any polygon is permitted to be wet, partially wet, or dry. Furthermore, combining Eqs. 3 and 6 leads to

$$\begin{aligned} H_j(\eta_j^n) &= \frac{1}{\lambda_j} \int_{s_j^1}^{s_j^2} \max[0, h(x(s), y(s)) + \eta_j^n] ds \\ &\geq \frac{1}{\lambda_j} \int_{s_j^1}^{s_j^2} [h(x(s), y(s)) + \eta_j^n] ds = \bar{H}_j^n. \end{aligned} \tag{8}$$

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 522

In linear or piecewise linear models, the computed \bar{H}_j^n always underestimates the correct edge-averaged water depth in partially flooded regions implying a reduced celerity and a larger bottom friction. On the other hand, the more accurate edge-averaged water depth $H_j(\eta_j^n)$, thanks to the use of subgrid details, allows a more gradual and realistic transition from wet to dry grid points with less energy dissipation. For tidal dynamics, two major consequences arise when using the subgrid technique: a faster tidal traveling wave, in both ebb and flood cycles, and a larger tidal range.

3 Grid generation

Performing simulations using the subgrid technique requires building models with a subgrid resolution. Smile Consult GmbH Hannover developed the Janet preprocessor for the generation of unstructured orthogonal grids with subgrid scale bathymetry that comply with the subgrid algorithm presented in Section 2. The objective was to investigate different grid generation strategies with subgrid technology for varying computational grid sizes and apply them to typical BAW applications. In this section, some important outlines of grid generation are presented.

Available topographic data are typically stored in a relational database management system. The database interface of the grid generator (preprocessor) can process different kinds of data sets. All data sets are described with metadata. The different interpolation strategies use a database-enabled spatial search engine. For every computational edge, a set of subedges, each with one length and one depth (depth classes of the subedges), is created. Similarly, for every computational polygon, a set of subpolygons, each with one area and one depth (depth classes of subpolygons), is created. The number of depth classes may vary for each polygon or edge.

For the subgrid generation, the following strategies were developed:

- Subdivision (SD) strategy with a uniform subdivision of all edges and polygons in subedges and subpolygons as shown in Fig. 2. The resolution of the subdivision is user-defined. The database-driven interpolation delivers the corresponding depth at subgrid level.
- Terracing topography (TT) strategy with nonuniform depth level-aligned subedges and subpolygons in the form of bathymetric “terraces” as shown in Fig. 3. The depth of the isobaths is user-defined. The generation of isobaths is performed on a temporarily interpolated raster (database interpolation).
- The generated subgrid is subject to some constraints for optimization and validation purposes:
- Computational cells with no or tiny subcell area (<1%) are filtered and removed from the grid.
- Flow separation between computational cells that are physically not connected should be insured everywhere.
- Edges with no subedges are given one subedge with a 0.0-m length (default).
- Inflow edges must have at least one subedge with sufficiently large lengths.
- Terrain data should strictly overlap inflow sides.
- For both subgrid generation strategies, the boundary fit is performed at a subgrid level.

Another important aspect during grid generation is the possible 1D discretization of tributaries with solely one computational polygon over the cross-section and subgrid bathymetry as illustrated in Fig. 4. Most of the aforementioned constraints are naturally taken into account in an unstructured grid including flow alignment for less numerical diffusion.

The previous techniques were successfully used to generate different structured and unstructured grid series for the Elbe Estuary with varying computational grid sizes but identical subgrid bathymetry. Grid variants of the Elbe Estuary were designed for systematic investigations with the new technique. In Section 5, a detailed description of one of the grids used in a real-world application is given.

4 Methodology

In this section, we will recall the basics of the semiimplicit solution algorithm and the associated numerical approximations used in UnTRIM² which refers to the new subgrid-based computational core (Casulli 2009). The solution algorithm remains very similar to the one used in the classic linear or piecewise linear UnTRIM (Casulli and Walters 2000). We shed also some light on the parametrizations of two important processes namely vertical turbulence and bottom friction. For the latter, a

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

new empirical correction is proposed. The solvers involved in the present work are the piecewise linear UnTRIM (designated by classical) and the nonlinear subgrid UnTRIM² (designated by sub-grid).

4.1 Numerical approximation

UnTRIM² solves the 3D, time-dependent, nonlinear differential equations related to hydrostatic and nonhydrostatic

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 523

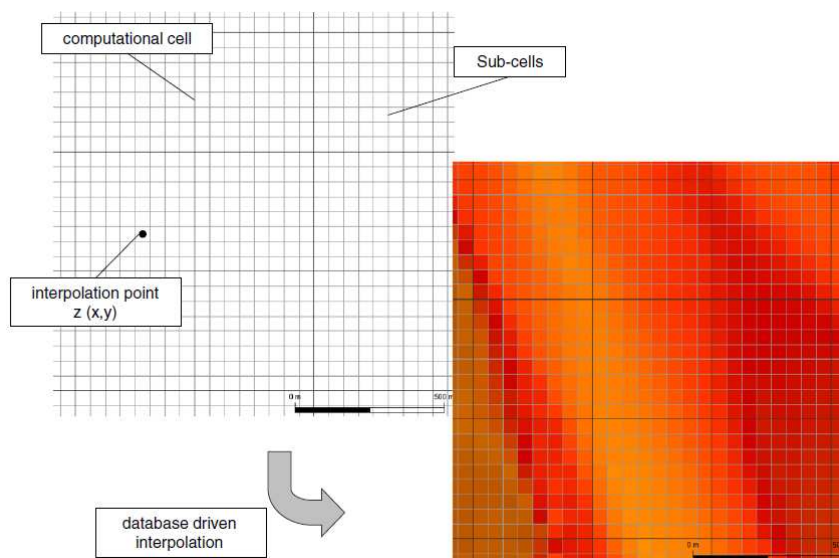


Fig. 2 Grid generation using the subdivision (SD) strategy. Left, computational grid and sub-grid interpolation; right, corresponding subgrid bathymetry

free-surface flow problems on an unstructured orthogonal grid to cover problems with complicated geometry. Numerically, UnTRIM² is based on a semi-implicit finite difference scheme that allows unconditional stability. Terms affecting stability like bottom friction, wind stress, and vertical mixing are treated implicitly. An Eulerian-Lagrangian explicit finite difference operator is used to account for the discretization of advection and horizontal dispersion. For stability, the implicitness factor θ should be taken in the range $0.5 \leq \theta \leq 1$. We refer to Casulli and Cattani (1994) and Casulli and Walters (2000) for a detailed description of the algorithm. Scalar transport processes are treated by an explicit mass conserving finite volume scheme with subcycling if necessary to ensure stability. The scheme allows higher accuracy through the use of flux limiters (Casulli and Zanolli

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

2005). In the case of baroclinic flow, the transport equations are coupled with the momentum equations through density gradients. The baroclinic forcing terms are

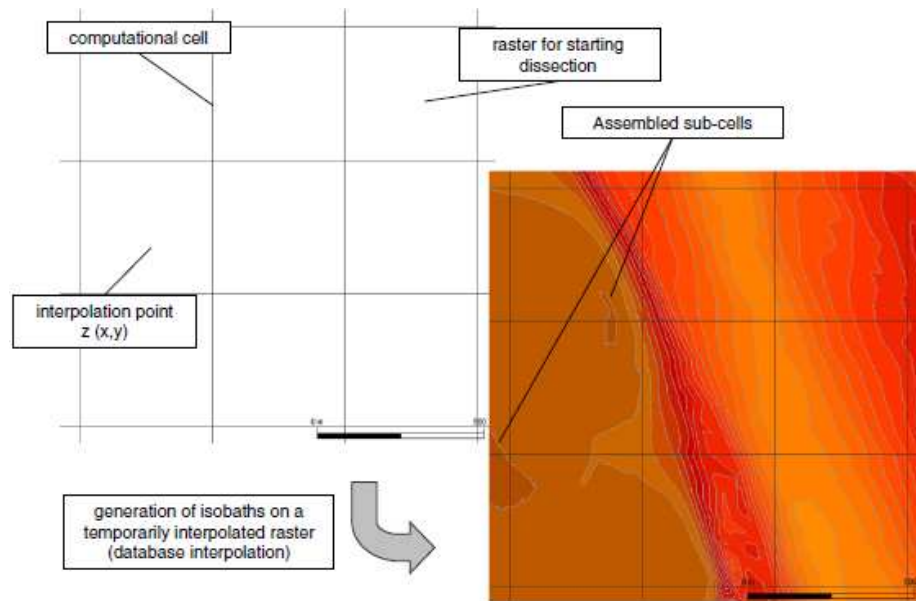


Fig. 3 Grid generation using the terracing topography (TT) strategy. Left, computational grid and auxiliary grid interpolation; right, corresponding subgrid bathymetry

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 524

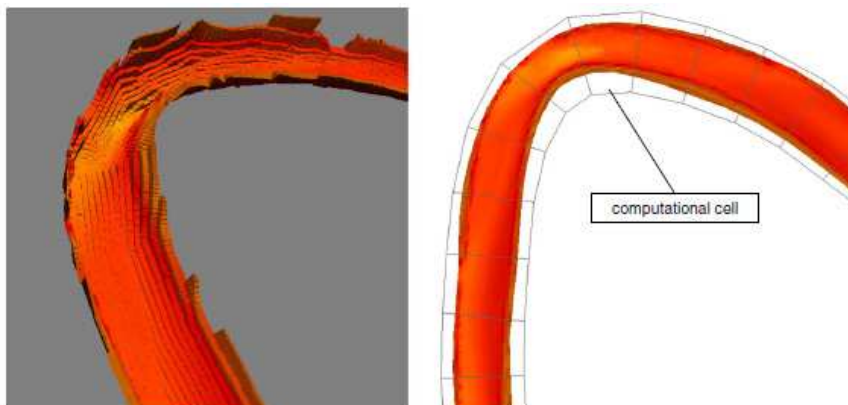


Fig. 4 Subgrid bathymetry and 1D discretization of a tributary river of the Elbe Estuary using the SD strategy. Left, 3D view of the subgrid bathymetry; right, computational grid with solely one computational cell over the cross section

solved explicitly in the momentum equations. The equations of transport are solved lagged one time step. In this case, the numerical scheme is subject to a weak Courant-Friedrichs-Lewy (CFL) stability condition. It is also subject to a weak stability condition due to the explicit treatment of horizontal diffusion in the momentum equations.

4.2 Turbulence model

Along with UnTRIM², the popular generic length-scale (GLS) turbulence model (Umlauf and Burchard 2003) was implemented. Following the Kolmogorov-Prandtl concept, eddy viscosity and eddy diffusivity are proportional to the square root of the turbulent kinetic energy and an integral turbulent length scale. The nondimensional proportionality coefficients are stability functions which describe the effect of shear and stratification (buoyancy). They are derived algebraically from the transport equations of the Reynolds stresses after parametrization.

The GLS solves two transport equations for the turbulent kinetic energy k and another turbulent quantity called ψ that is used to establish the turbulence length scale in a generic way. Through the specific choice of some generic parameters, the quantity ψ obtains a specific physical meaning. The implementation is able to recover existing turbulence closure models like k - ϵ , k - ω , or k - kl or develop new models like the generic k -gen closure suggested by Umlauf and Burchard (2003). In our simulations, the k -closure model where ψ stands for energy dissipation was used along with the Kanta-Clayson stability function. The transport equation of the two quantities k and ψ are solved in the 1D vertical staggered grid extending from the bottom to the free surface with von Neumann boundary conditions. The turbulent variables of the turbulence model defined at cell vertical interfaces naturally fit to the staggered grid of UnTRIM, and no further interpolation is required. In flows with density gradients like during salt intrusion in estuaries, the two equation model, which explicitly takes into account buoyancy, shows a more realistic vertical stratification when compared to algebraic zero-equation closure models with stability functions based on the Richardson number. An example for salinity stratification will be shown in Section 6.

4.3 Bottom friction

A highly resolved bathymetry at subgrid level with the assumption of a constant velocity along a computational edge requires a modified treatment of processes involved at the lowest layer of the model particularly bottom friction. It is well known from cross-section-integrated models that the combination of the cross-sectional mean velocity with a constant bottom friction parameter tends to systematically overestimate the energy dissipation in shallow regions leading to higher flow resistance. This problem can be mitigated by the use of the conveyance approach (Stelling et al. 1998) which allows the computation of a uniform and consistent cross-sectional energy dissipation. Unfortunately, the conveyance approach only works for 2D vertically averaged models. Some preliminary meaningful experiences conducted on simple geometries like a U-shaped channel (see Fig. 5) showed that there is a systematic overestimation of bottom friction particularly when the resolution of the computational grid becomes low.

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

In these basic experiments, different situations with a steady flow have been studied in order to figure out the sensitivity of the results against bottom friction and the dimension (2D, 3D) of the model. All grids share the same (identical) bathymetry shown in Fig. 5 (left) which is given on a subgrid scale. As sketched in Fig. 5 (right), starting from the fully resolved classical representation (CG010m, reference), the cross-sectional resolution of the

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 525

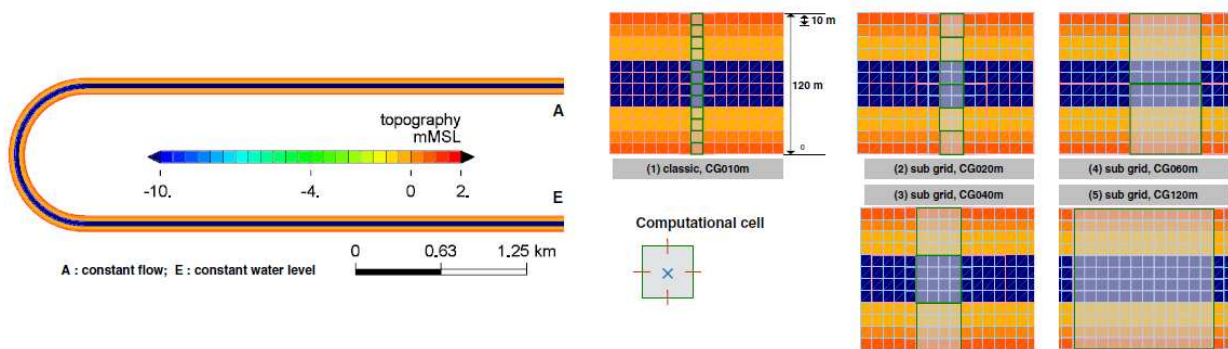


Fig. 5 Bathymetry of the U-channel (left) and five grid schematization (right). Computational grid (CG) versus subgrid (SG)

computational grid is gradually coarsened until a crosssection-integrated model (CG120m). In grids CG020m and CG040m, bathymetry varies moderately within a polygon. Grids CG060m and CG120m, however, exhibit a strongly varying bathymetry within a computational polygon. All five models were run under identical parametrization and boundary conditions. The variation of water level along the channel is displayed in Fig. 6 (left) for the five grids shown before. Two groups of curves are displayed: the group with the finer grids exhibits a water level gradient that is identical to the classical reference solution (1). The water level gradient obtained for the coarser grids, where bathymetry varies significantly within a computational grid cell, is larger than that for the finer grids. The latter is essentially due to the overestimation of bottom friction when using constant cross-sectional velocity in a large computational cell. Based on the conveyance approach, an appropriate empirical correction was worked out.

Thus, subgrid velocities are calculated from the mean edge velocity for every specified layer z . The correction applies for both depth-averaged and vertically structured models and requires the introduction of a model-dependent factor p :

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

$$\frac{u_{j,k}(y)}{u_{j,k}} = \frac{1}{f_{j,k}} \left[\frac{H_{j,k}(y)}{\gamma_j(y)} \right]^p, \quad (9)$$

$$f_{j,k} = \frac{1}{A_{j,k}} \int_{\Gamma_j} \Delta z_{j,k}(y) \left[\frac{H_{j,k}(y)}{\gamma_j(y)} \right]^p dy \quad (10)$$

If bottom friction correction is applied as shown above, we can see that the water level gradient in the 2D case becomes now almost independent of grid resolution (Fig. 7 left). If we step from 2D to 3D, the results are still satisfactory as shown in Fig. 7 (right). The factor p is a calibration parameter ranging between 0 and 1 designed in such a way that for

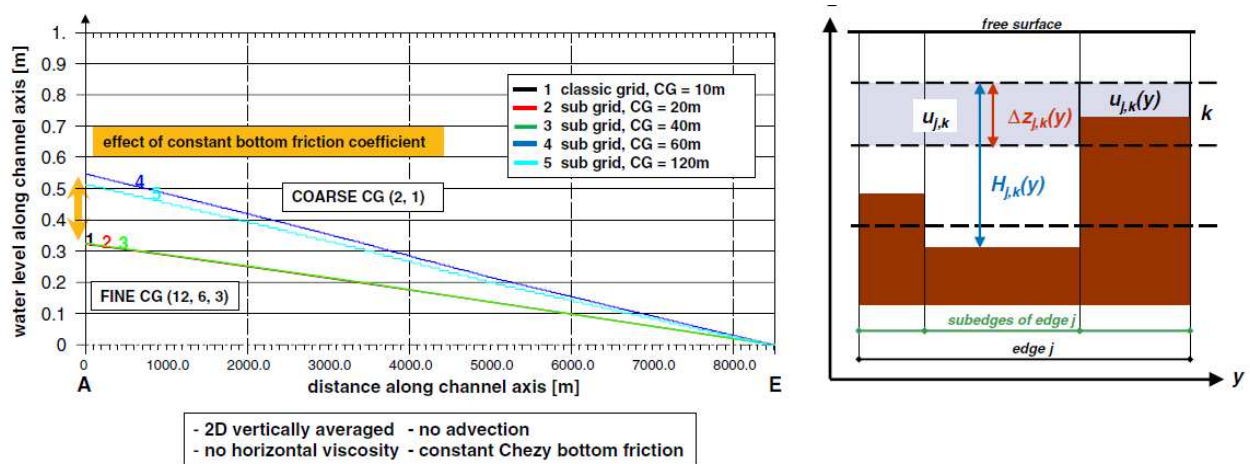


Fig. 6 Results for the U-channel 2D simulation with Chezy bottom friction and no advection (left). Schematization of the subgrid velocity $u_{j,k}(y)$ and the layer-averaged velocity $u_{j,k}$ (right)

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 526

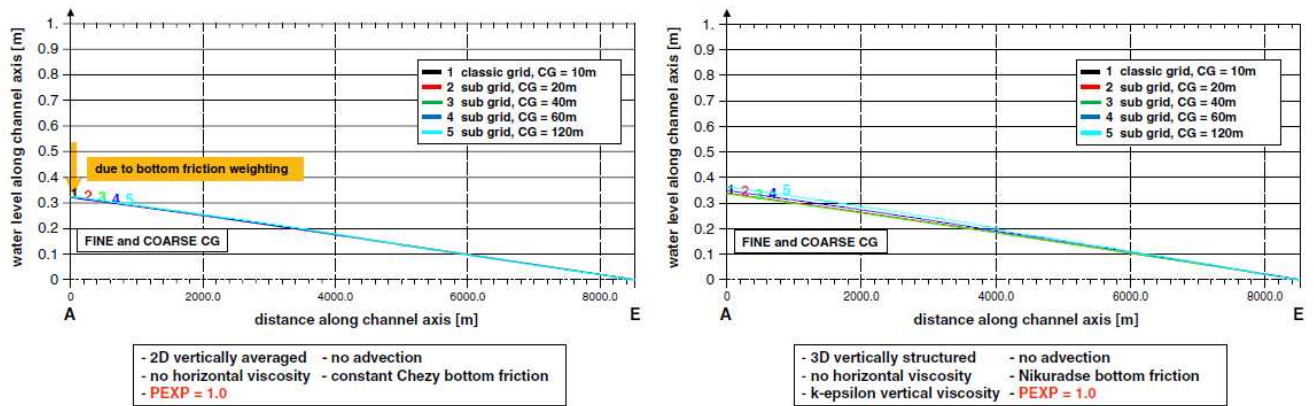


Fig. 7 Results for the U-channel 2D (left) and 3D (right) simulations. Calibration factor $p_{exp} = 1.0$

$p = 0$, or if no subgrid variation of the bathymetry is considered (constant bathymetry along edges), the computation of bottom friction reduces to the original formulation.

5 Examples of subgrid models

Three major estuaries are located at the German North Sea coast: Elbe, Weser, and Ems. In this study, we have focused on the Elbe Estuary. In addition to a classical grid, a series of computational grids using subgrid-scale bathymetry were generated:

- UG400SD24: based on the SD strategy with a computational unstructured grid having a 400-m average resolution, 24 subdivisions along the edges, and 24×24 subdivisions inside polygons.
- UG200SD12: based on the SD strategy with a computational unstructured grid having a 200-m average resolution, 12 subdivisions along the edges, and 12×12 subdivisions inside polygons.
- UG200SD06: based on the SD strategy with a computational unstructured grid having a 100-m average resolution, six subdivisions along the edges, and 6×6 subdivisions inside polygons.
- UG050SD03: based on the SD strategy with a computational unstructured grid having a 50-m average resolution, three subdivisions along the edges and 3×3 subdivisions inside polygons.

These grids were generated at the starting phase during the implementation of the UnTRIM² subgrid technique at BAW. The aim was to settle the kind of computational grids that should be considered (structured or unstructured) as well as to figure out the extent to which the resolution of the computational grids can be coarsened. The results were compared to the classical unstructured model which stands for the reference. An extensive work was done in this direction including the introduction of the bottom friction correction as explained in Section 4.3. The outcome of this preliminary study was that the sensitivity on computational resolution was mild to moderate and that

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

even the coarsest computational grid (UG400SD24) was still able to deliver satisfactory results as long as bottom friction correction is switched on and a suitable calibration factor is used (see Section 4.3). The trade-off between numerical efficiency and accuracy clearly put the coarsest grid forward for potential full-scale simulations like in the case of the Elbe Estuary model. Thus, we will restrict ourselves to the coarsest UG400SD24 model depicted in Fig. 8. For comparison, a typical classical unstructured grid is used. Characteristics of both models are presented in Section 6. Figure 9 highlights a section of the Elbe Estuary at the junction with the tributary river Str. It gives an impression of how differently bathymetric details are represented in a classical grid and a grid using the subgrid technique. Moreover, the tributary river is taken into account in a 1D way. It was neglected in the classical grid due to numerical constraints. On the second closeup area shown in Fig. 10, one can clearly see that the bathymetry of the tidal flats can be resolved even better in a coarse computational grid model using subgrid-scale bathymetry (UG400SD24) compared to a highly resolved classical grid model.

With the subgrid technique, the simulation of flooding and drying in these areas can be performed more realistically. For any water level, an excellent approximation of the water volume as well as the active cross section for the flow is guaranteed.

Narrow harbor channels and basins do also represent a challenge for traditional grid generation. Geometries of these regions are often complex, requiring a particular attention during meshing. The elaboration of such grids remains always a fussy business mainly due to the need of smooth transitions between channels of diffe-

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 527

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

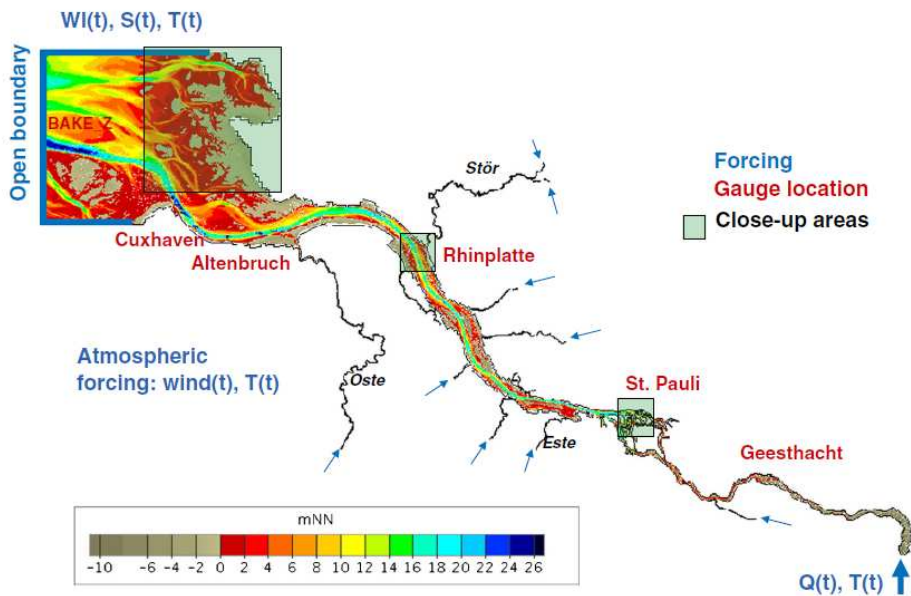
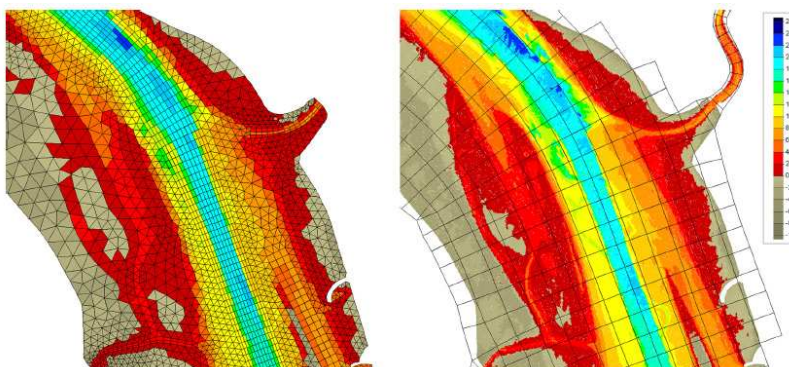


Fig. 8 Bathymetry of the Elbe Estuary with the UG400SD24 model (400-m average computational grid resolution, 24 subedges per edge, and 24×24 subpolygons per polygon). Locations of tidal gauges, discharges, and closeup areas are depicted as well

rent sizes. In some hydrodynamical systems, this problem is bypassed by the replacement of the detailed harbor branching with simple geometries containing the relevant volume of water. With subgrid technique, harbor channels and basins are realistically and accurately represented at the subgrid level including boundary fit as shown in Fig. 11. The harbor area can be covered using a coarse computational grid as long as one is not interested in local details of flow, e.g., flow separation near harbor basin entrances.

6 Application

In this section, one application of the subgrid technique is presented. The model covers the Elbe Estuary and focuses



Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

Fig. 9 Bathymetry at the junction of the Elbe Estuary with the Stör tributary using classical unstructured grid (left) and subgrid technique (right)

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 528

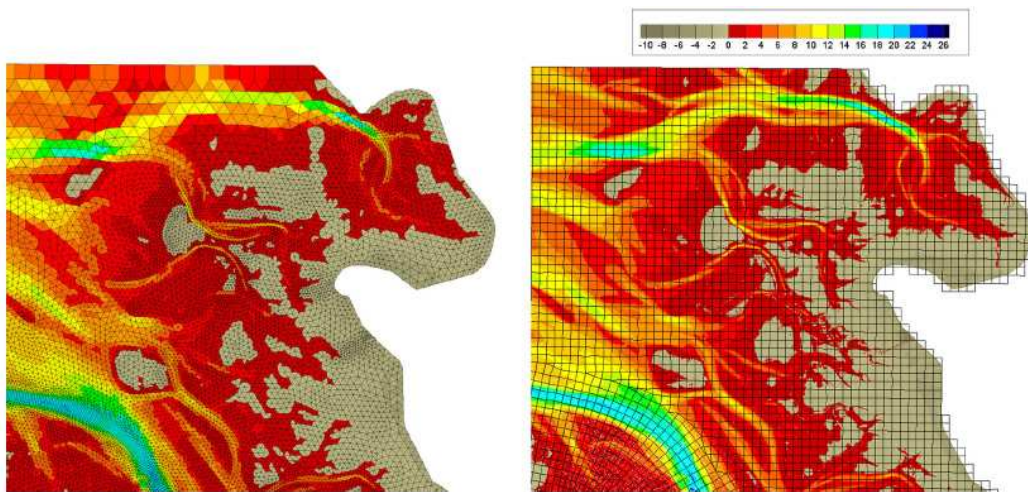


Fig. 10 Bathymetry of tidal flats with classical unstructured grid (left) and subgrid technique (right)

on operational water level, salinity, and temperature forecast. The model was run first in hindcast simulations for calibration and validation purposes.

6.1 The Elbe Estuary operational model

The Elbe Estuary is a very important waterway. Ship traffic requests precise and preferably long-term water level forecasts. Particular events like storm surges and periods of low or high freshwater discharge have great economic significance and require therefore detailed temporal and spatial forecasts for water level, current velocity, salinity, and eventually temperature in the case of ice sheet formation.

In the Elbe, the weir near Geesthacht (see Fig. 8) represents the artificial limit between the river and the estuary. The river feeds the estuary with time-varying freshwater runoff. The seaside boundary at the North Sea is influenced by tidal dynamics, waves, external surges, and storm surges. During the flood tide, salt water with higher density intrudes into the estuary. By means of hydrodynamic models, the variation in water level, current, salinity, and temperature can be simulated with reasonable accuracy. In order to accurately represent the progression of the tidal wave, an appropriate grid resolution on top of a prevailing topography is required. This is the scope

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

of the operational model for the Elbe Estuary we present in this section. The numerical model for water level, salinity, and temperature of the Elbe Estuary at BAW was first developed using the classical highly resolved unstructured piecewise linear UnTRIM. Later, the classical model was replaced



Fig. 11 Bathymetry of Hamburg harbor using classical unstructured grid (left) and subgrid technique (right)

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 529

by UnTRIM² (Casulli 2009) based on the subgrid technique. The forcing is identical for both models. The daily meteorological forcing data (wind and temperature) are delivered by the German Weather Service (DWD) using the COSMO-EU model (Doms 2011) which covers almost the whole Europe including the Baltic Sea, the Mediterranean Sea, and the Black Sea with 665×657 grid points in a 0.0625° (~7 km) resolution. Open boundary data are delivered by the German Maritime and Hydrographic Agency (BSH) using the coastal model HIROMB-BOOS-MODEL (HBM) (Berg and Weismann Poulsen 2012) which is the successor of the previous BSHmod model. The operational service at BSH is based on a 3D baroclinic circulation model run on a North Sea/Baltic Sea grid with a horizontal resolution of 3 nautical miles and a fully two-way nested 0.5-nautical-mile grid of the German Bight/Western Baltic.

In Fig. 12, a flow chart of the model operation is sketched. The meteorological model delivers ground air temperature and zonal and meridional wind velocities at a 10-m height. The coarse forecast data of the meteorological forcing are processed by the PREMTEO preprocessor which performs an interpolation from the meteorological grid (7 km × 7 km) onto the UnTRIM²

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

computational grid. The coarse resolution of the meteorological grid leads to an underestimation of wind velocities in grid cells that are partially wet like, for instance, in narrow parts of the estuary. Therefore, the so-called Wind Atlas Analysis and Application Program (WASP) factors (Mortensen et al. 1993) were introduced by DWD to enhance the wind fields over the estuary. Along with a fine WASP grid (250 m × 250 m) and 12 wind direction parameters (each every 30°), WASP correction factors are computed and multiplied with the interpolated meteorological data.

Heat exchange at the air-water interface is parameterized by an Excess Temperature Model (Sweers 1976) where only the background temperature of air is required. Thereby, the heat exchange flux in joules per square meter per second per degree Celsius is computed using a bulk exchange formula. At the open boundary, the forecasted data of water level, temperature, and salinity delivered by the HBM model are processed by the PREHYDRO preprocessor to fit with the UnTRIM² open boundary. For salinity and temperature at the open boundary, vertically averaged values are used. At the inflow boundary, measured freshwater discharge from the Federal Institute of Hydrology (BfG) at the gauge Neu Darchau is used. Water temperatures at inflow are daily measurements at gauge Cumlosen delivered by the River Basin Community Elbe (FGG Elbe).

A typical forecast simulation is performed for 24 h. Comparison is performed between the coarse computational grid model with subgrid bathymetry (UG400SD24) and the classical model.

6.1.1 Results

Obviously, the quality of the forecasts delivered by the estuary operational model depends strongly on the accuracy of the forcing at boundaries, in particular water levels predicted by the HBM model at the North Sea boundary. A measurement gauge (Bake Z) located right on the open boundary (see Fig. 8) allows to assess the quality of such predictions. Discrepancies at open boundaries will automatically be transferred along the estuary. Therefore, the model was first calibrated and validated for different scenarios using measured water level at the seaside boundary. The main calibration parameters that were investigated are bottom friction and turbulence (see Section 4).

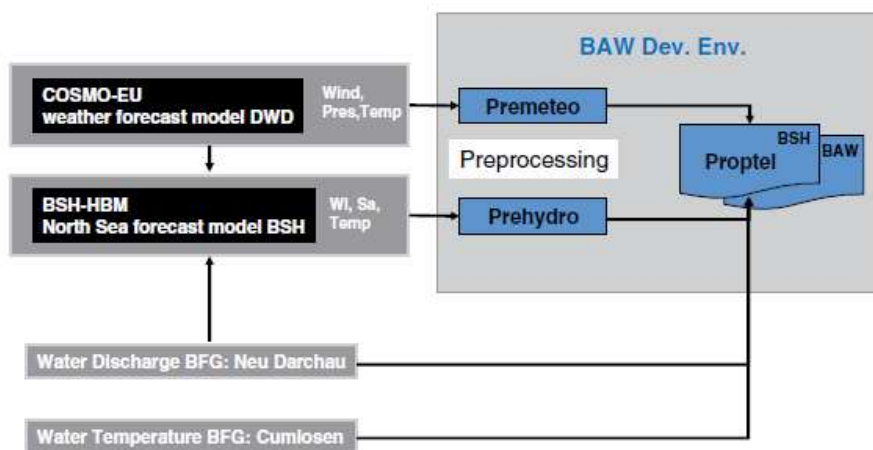


Fig. 12 Flow chart for boundary data processing. The left part highlights the origin of the forcing data at boundaries. The right part represents the forecast model PROPTEL

Calibration and validation Apart from mean conditions, the year 2006 involved various particular events including a storm surge and very high/very low discharges and was hence chosen for the calibration and validation processes. Figure 13 shows results for a period of 11 days out of a 14-day (22 October 2006–4 November 2006) model run. The results highlight water level predictions before, during, and after the occurrence of a storm surge. They were obtained by a 3D simulation including salinity and temperature. The quality of the results as well as the numerical performance of classical and subgrid models are discussed here. The water level at the St. Pauli gauge (see Fig. 8) is an important benchmark for model results. In the upper part of Fig. 13, both tidal signal and tidally averaged (over 12 h and 25 min) water levels are in good agreement with measurement for classical and subgrid models. If we look at the differences between model results and measurements in the lower part, we see comparable discrepancies independently of the model used. In some parts, the classical model is better; in other parts, the subgrid model shows better agreement with measurements. However, the major difference between the two models resides in the computational efficiency. Table 1 summarizes some characteristics and performance of both models. The classical model spends 5 h and 30 min to perform 14 days of simulation, whereas the subgrid model needs only 22 min using the same numerical settings and the same computational resources (16 cpus of

	Classical	Subgrid
No. of edges (subgrid)	194,453	28,369 (293,680)
No. of polygons (subgrid)	120,124	12,393 (584,609)
vertical resolution (m)	1	1
Time step (s)	100	300
Real cpu time (min)	330	22
Speedup vs real time	61	916

Table 1 Characteristics and computational performance of classical and subgrid models for a 14-day simulation

one HPC-System node). This represents roughly a speedup factor of 15.

Operational mode In this study, the period January to March 2011, which covers amongst others both a storm surge and a high discharge event in the Elbe Estuary, was investigated. The simulation period with the corresponding measured freshwater discharge at inflow boundary as well as the particular events are shown in Fig. 14 and Table 2. The models in operational mode are now driven with forecasted forcing data. Figures 15, 16, and 17 show results for three different scenarios at the St. Pauli gauge together with Bake Z. Again, a very good agreement between classical and subgrid models is obtained for all three scenarios. The subgrid model, however, has a slightly higher tidal range

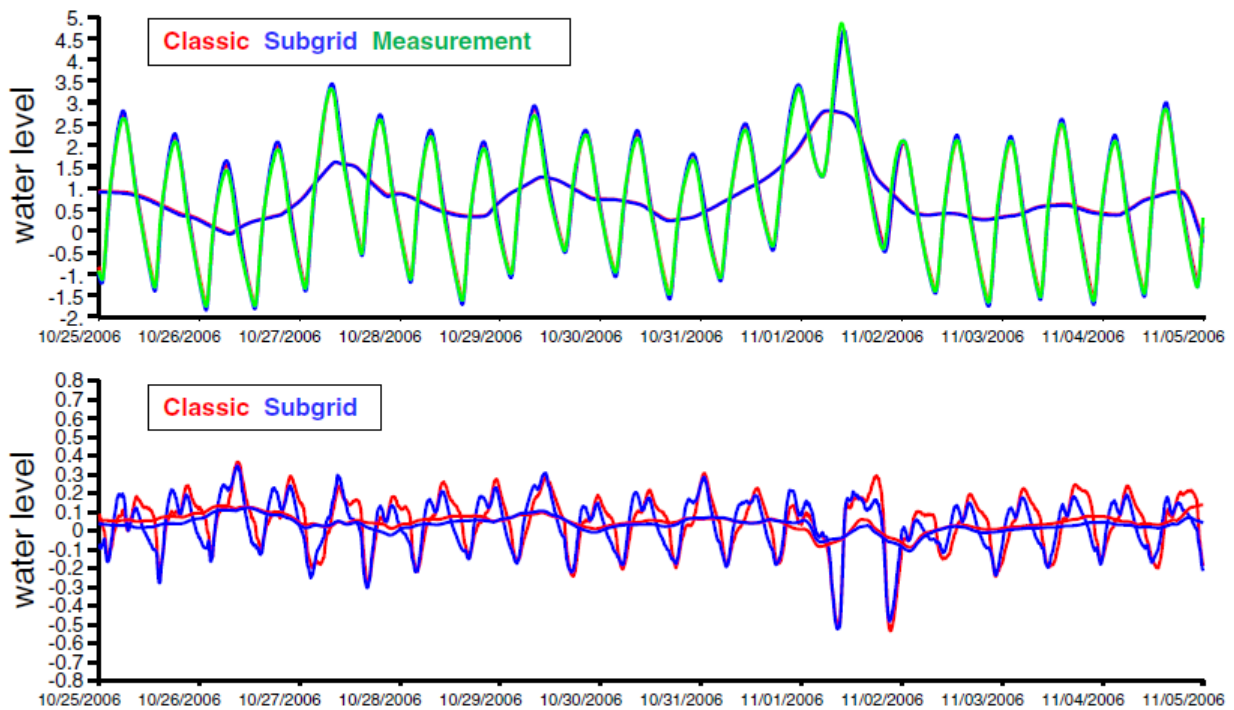


Fig. 13 Water level at Hamburg St. Pauli for 14 days of simulation. Top, comparison between measurements (green), classical (red), and subgrid (blue) results; bottom, differences between simulations and measurement. The second set of curves are tidally averaged results (over 12 h and 25 min)

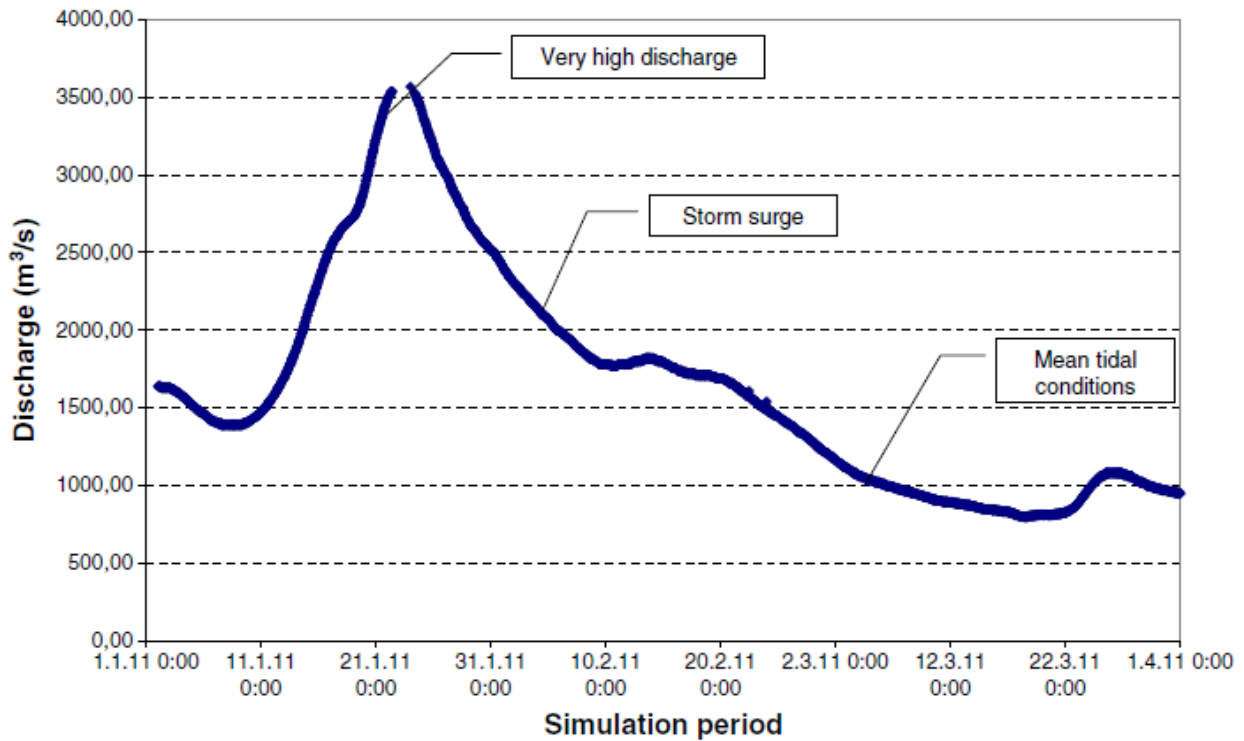


Fig. 14 Discharge of the river Elbe during the first quarter of 2011. Data were provided by BfG for Neu Darchau, 10 km upstream of Bleckede, the inflow boundary of the model. Special events during the simulation are highlighted as well

which is principally due to lower low water. This is most likely a direct consequence of its accurate representation of tidal volume through the use of highly resolved subgrid bathymetry. However, it is very important to bear in mind that the quality of these results depends very strongly on the boundary forcing delivered by the HBM model as shown on the curves at gauge Bake Z. Every deviation between measured and predicted data at the open boundary will be automatically reflected on all other locations along the estuary. The substantial benefit of the subgrid technique arises again in the computational performance. A typical 24-h operational forecast using the classical model requires an average of 100 min. For the same setting, the subgrid model needs only 5 min using the same computational resources (8 cpus on local 8-core PC). The computational speedup compared to the classical model is in the order of 20. Compared to the calibration simulation, there is a clear decline in computational speedup versus real process time. This is particularly due to meteorology preprocessing and to extensive IO operations. In the operational mode, the preprocessor PREMTEO requires a nonnegligible time for performing interpolation of meteorological data from the coarse COSMO-EU grid onto the UnTRIM computational grid including some data copying from remote servers (Table 3).

Scenario	Period	Event
1	5–9 March 2011	Mean tidal conditions
2	19–24 January 2011	High water discharge
3	3–8 February 2011	Storm surge

Table 2 Scenarios for comparison between classical and subgrid models in operational mode

The substantial gain in computational speed with the subgrid technique allows us to perform some extra calculations involving other boundary conditions and parameterizations to improve results.

Looking at depth-averaged spatial distribution of current velocities, we do observe in the case of subgrid simulations no loss of bathymetric detail but loss of small-scale flow variations due to the lower resolution of the computational results. Velocity magnitude and current vectors show comparable patterns for both subgrid and classical models as seen in Fig. 18. Nevertheless, due to the fact that the velocities obtained from the subgrid model represent mean values on a relatively coarse computational grid, the maximum velocities are larger on the highly resolving classical grid.

Salt transport processes in the estuary are affected by tidal current, freshwater discharge, turbulent mixing, and baroclinic circulation. Therefore, salinity represents an appropriate tracer for the assessment and calibration of scalar transport. The vertically averaged spatial distribution as well as the vertical profile of salinity along the Elbe shown in Figs. 19 and 20 respectively indicate a deeper salt intrusion into the estuary for the subgrid model. This observation is confirmed in Fig. 21 by time series of salinity at different measurement locations where a systematic overestimation of salinity by the subgrid model is observed. The difference ranges between 1 and 2 psu.

7 Conclusions

The subgrid technique offers the possibility to take into account the bathymetric information at the subgrid scale allowing model bathymetry to be accurate up to

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

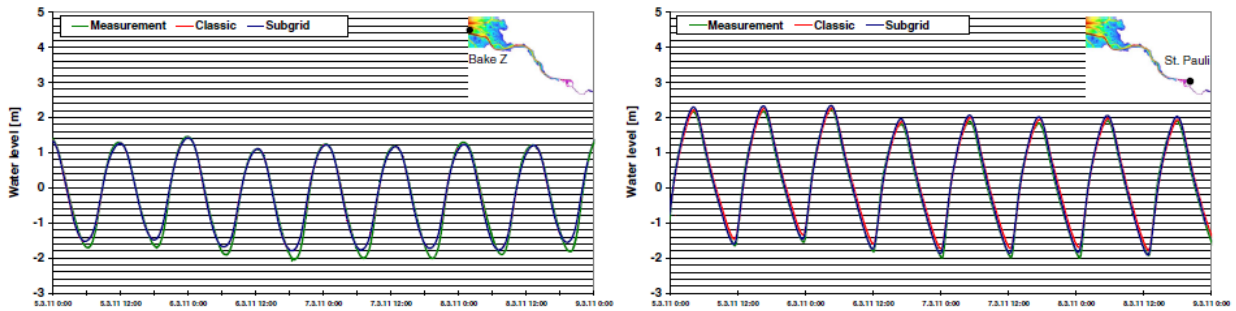


Fig. 15 Water level for mean tidal conditions near open boundary at Bake Z (left) and for Hamburg harbor in St. Pauli (right). Comparison is shown between classical model (red), subgrid model (blue), and measurements (green) for the period 5–8 March 2011

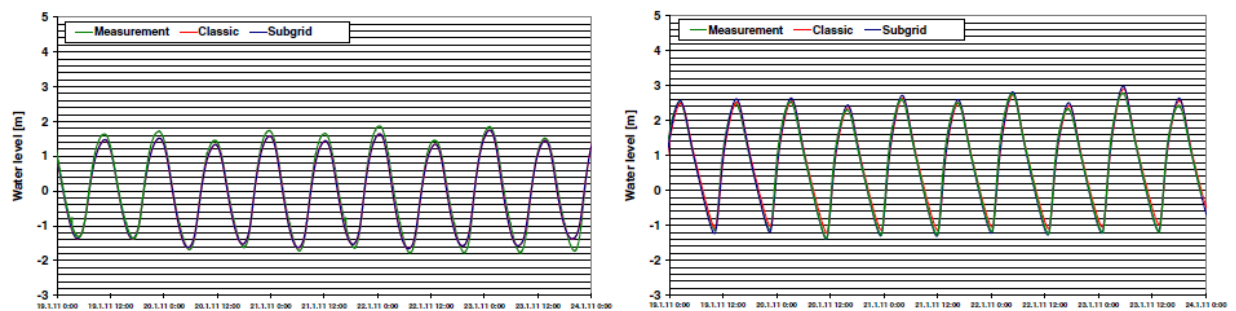


Fig. 16 Water level during high discharge near open boundary at Bake Z (left) and at Hamburgs harbor in St. Pauli (right). Comparison is shown between classical model (red), subgrid model (blue), and measurements (green) for the period 19–23 January 2011

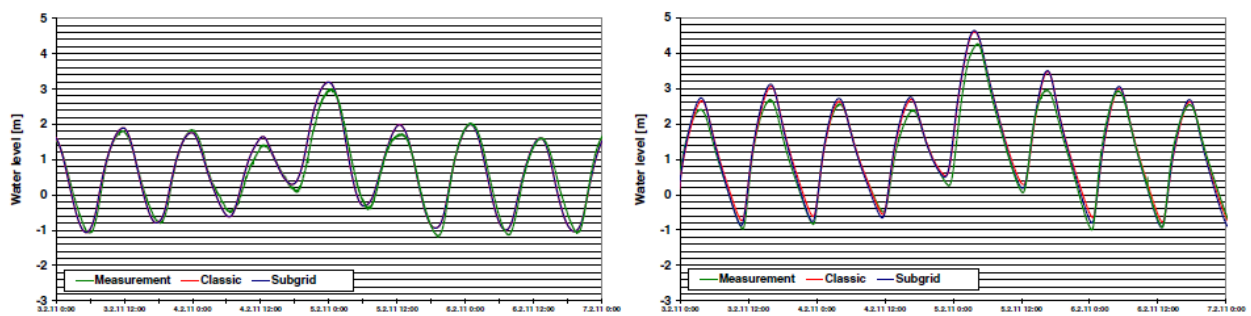


Fig. 17 Water level during storm surge near open boundary at Bake Z (left) and at Hamburgs harbor in St. Pauli (left). Comparison is shown between classical model (red), subgrid model (blue), and measurements (green) for the period 3–6 February 2011

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

Time step (s)	100	300
Real cpu time (s)	6000	300
Speedup vs real time	14	288

Table 3 Computational performance of classical and subgrid operational models for a 24-h forecast simulation

measurement accuracy independent from the resolution of the computational grid. The subgrid technique shows less restrictions on grid generation: e.g., easy fit of lateral boundaries, potential 1D approximation of tributary rivers, use of homogeneous, flow-aligned unstructured grid (quadrilaterals, triangles if needed), and an accurate representation of flow area and volume for every water level. Consequently, simulation of wetting and drying of tidal flats can be performed more accurately.

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 533

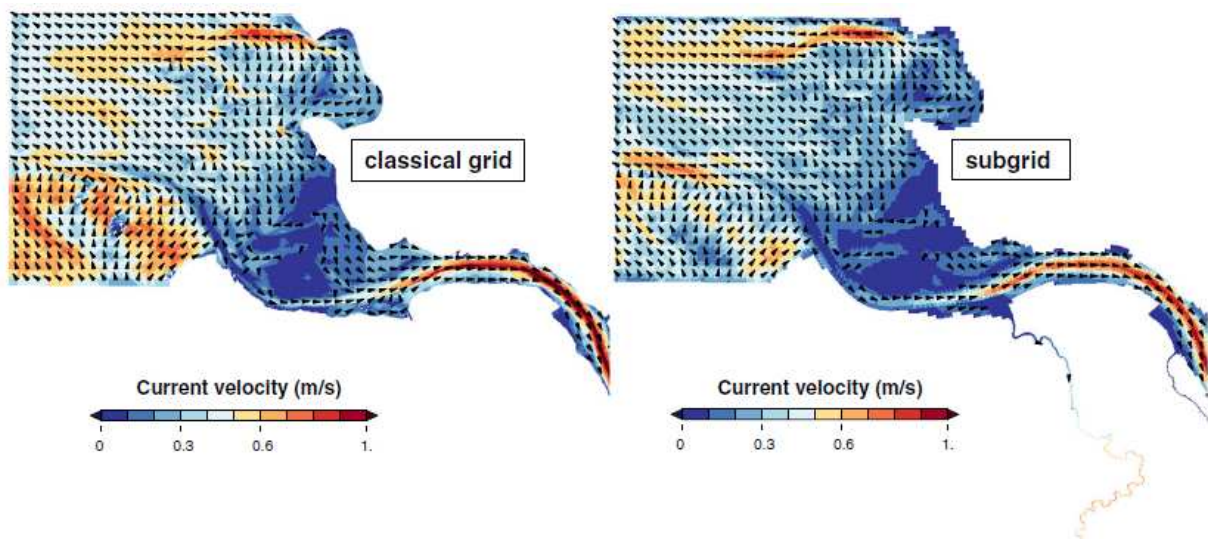


Fig. 18 Vertically averaged current velocity at the mouth of the Elbe Estuary on 5 February 2011 at 2:05 am during a storm surge. Flood current prevails in the narrow upstream part of the estuary, while ebb current dominates in the wide seaward part. Comparison is shown between classical model (left) and subgrid model (right)

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

Comparison of simulation results for the Elbe Estuary obtained with a model based on the subgrid technique and a classical model showed good agreement for both long-term simulation of tidal dynamics and short-term operational forecasts. The computational speedup reached through the use of subgrid was about a factor of 20.

Although the time step used in the subgrid model is three times larger than the one used in the classical model, the fraction of elements where the CFL numbers are low ($CFL < 0.5$) is, in average, larger in the case of subgrid. The fraction of elements where the CFL numbers are high ($CFL > 1$), however, is in average lower in the subgrid case. As a result, if we want to obtain comparable external Courant numbers to the classical model, there is still potential to use larger time steps in the subgrid model, which means that we can go even faster. In other words, the resolution of the computational grid can be coarsened. On the subgrid level, we are using an

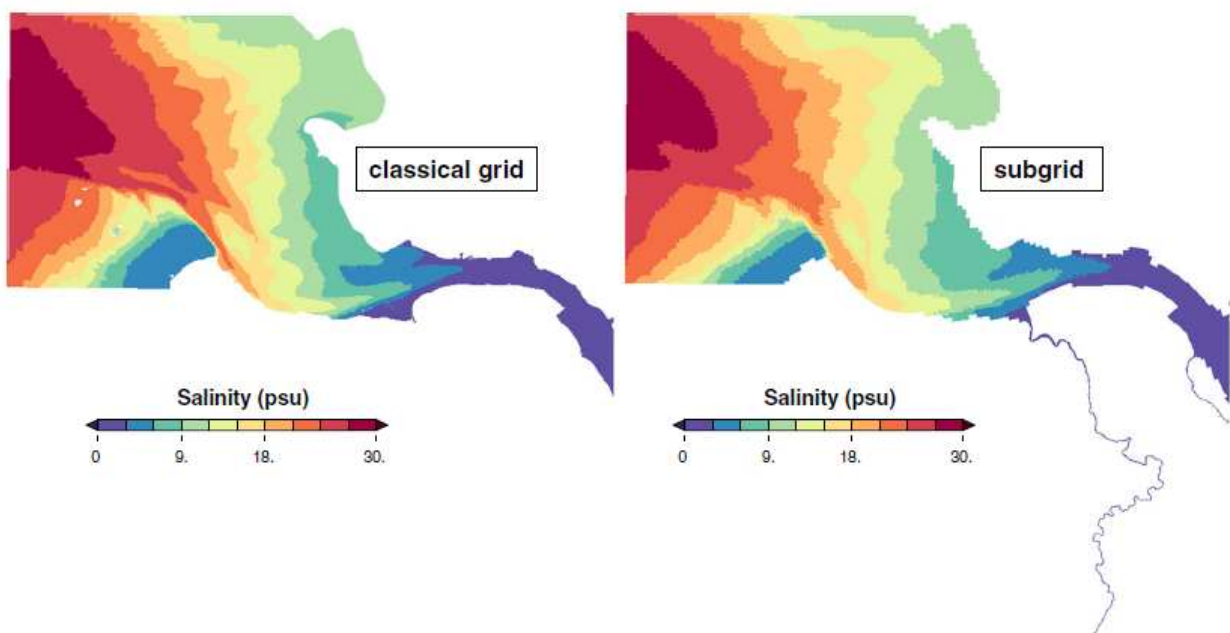


Fig. 19 Vertically averaged salinity at the mouth of the estuary on 5 February 2011 at 2:05 am during a storm surge (see Fig. 18 for depth averaged current velocity). Comparison is shown between classical model (left) and subgrid model (right)

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

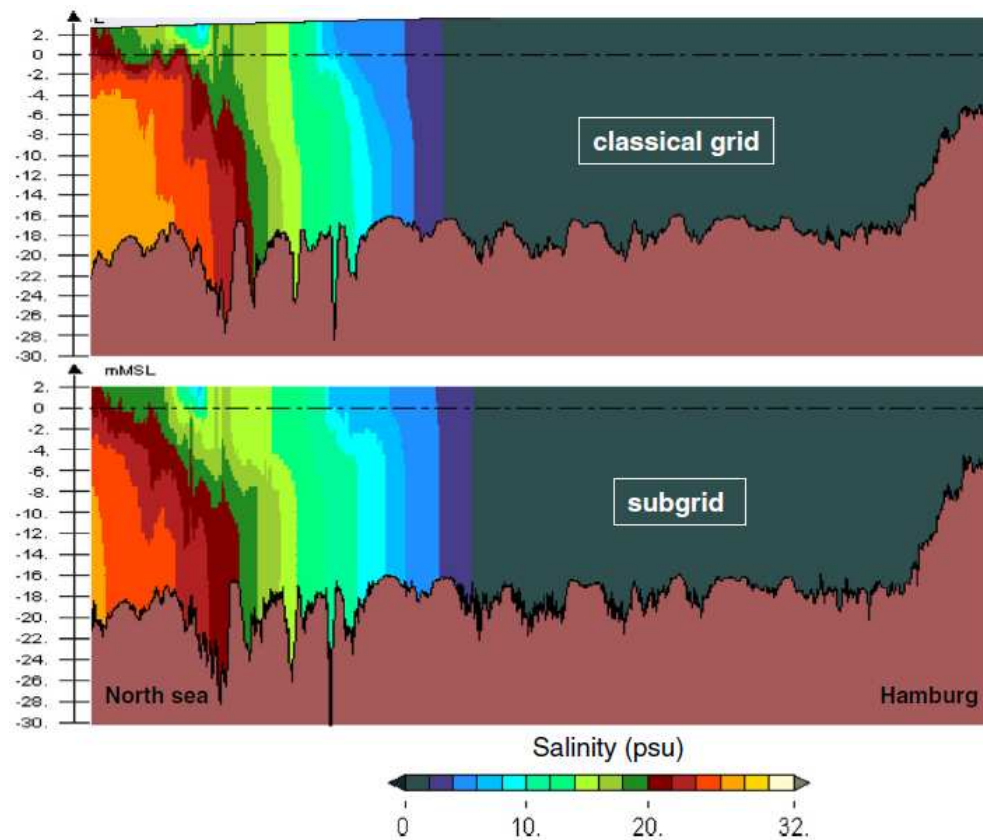


Fig. 20 Longitudinal section of salinity along the Elbe Estuary on 5 February 2011 at 2:05 am during a storm surge. Comparison of salt intrusion and vertical stratification between classical model (top) and subgrid model (bottom)

average resolution of 16 m which is for the investigated coastal domain with large intertidal mudflats sufficient. For other types of models, e.g., where the resolution of flow gradients is of significance, the subgrid resolution should be higher or even go to the limit of the available bathymetric data.

The subgrid technique improves accuracy if used with the same (high) resolution classical computational grid. On the other hand, comparable results are obtained if the subgrid technique is used even on a much coarser computational grid. Therefore, subgrid models are particularly suitable for simulations where real computation time is an important issue.

More generally, the subgrid technique is a promising framework to perform accurate temporal and spatial large-scale simulations of coastal and estuarine flows and transport processes like flooding and drying of large areas at low computational cost. Other potential applications of the subgrid technique are sensitivity studies of computational results with regard to model resolution.

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

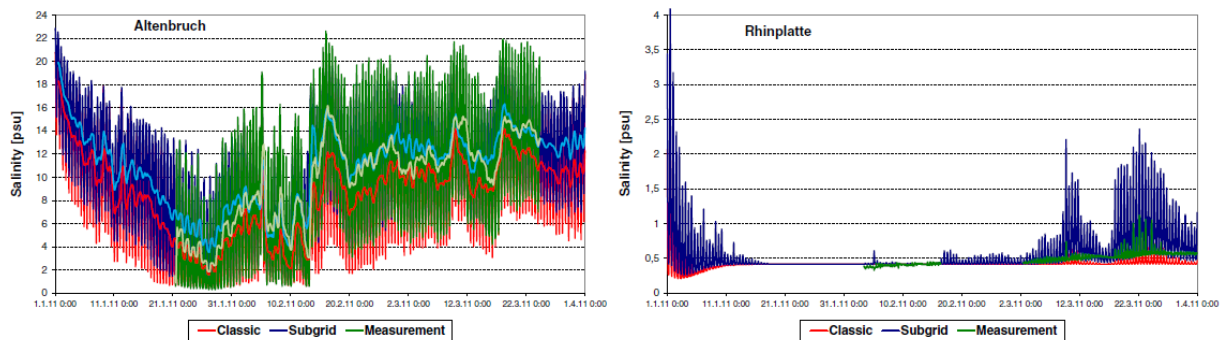


Fig. 21 Near-bottom salinity at Rhinplatte and Altenbruch for 3 months of simulation. Comparison is shown between classical model (red), subgrid model (blue), and measurements (green). Left figure (Altenbruch) exhibits also comparison of tidally averaged salinity between both models and measurements with corresponding colors

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation
Ocean Dynamics (2014), S. 519-535.

S. 535

Acknowledgments

The authors wish to thank the colleagues from BSH Hamburg for providing the necessary simulation forcing data.

References

- Balzano A (1998) Evaluation of methods for numerical simulation of wetting and drying in shallow water flow models. *Coast Eng* 34(83):107
- Bates PD (2000) Development and testing of a sub-grid-scale model for moving boundary hydrodynamic problems in shallow water. *Hydrol Process* 14(2073):2088
- Bates PD, De Roo APJ (2000) A simple raster-based model for flood inundation simulation. *J Hydrol* 236(54):77
- Berg P, Weismann Poulsen J (2012) Implementation details for HBM. DMI, Copenhagen. DMI technical report no. 12-11. www.dmi.dk/dmi/tr12-11.pdf
- Casulli V (1990) Semi-implicit finite difference methods for the twodimensional shallow water equations. *J Comput Phys* 86(56):74

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

- Casulli V (2009) A high-resolution wetting and drying algorithm for free-surface hydrodynamics. *Int J Numer Method Fluids* 60:391– 408
- Casulli V, Cattani E (1994) Stability, accuracy and efficiency of a semi-implicit method for three-dimensional shallow water flow. *Comput Math Appl* 27(99):112
- Casulli V, Cheng RT (1992) Semi-implicit finite difference methods for three-dimensional shallow water flow. *Int J Numer Methods Fluids* 15(629):648
- Casulli V, Stelling GS (2010) Semi-implicit subgrid modelling of three-dimensional free-surface flows. *Int J Numer Method Fluids* 67:441–449
- Casulli V, Walters RA (2000) An unstructured grid, three-dimensional model based on the shallow water equations. *Int J Numer Methods Fluids* 32(331):348
- Casulli V, Zanolli P (2005) High resolution methods for multidimensional advection-diffusion problems in free surface hydrodynamics. *Ocean Model* 10(137):151
- Doms GA (2011) Description of the non-hydrostatic regional COSMO model. COSMO Consortium. (<http://www.cosmomodel.org>). Accessed Sept 2011
- Horritt DM, Bates PD (2001) Effects of spatial resolution on raster based models of flood flow. *J Hydrol* 253(239):249
- Mortensen NG, Landberg L, Troen I, Petersen EL (1993) Wind atlas analysis, and application program (WASP). User guide, vol 2. Risø National Laboratory Roskilde, Denmark, p 133
- Stelling GS (2012) Quadtree flood simulations with sub-grid digital elevation models. In: *Proceedings of the Institution Of Civil Engineers ICE*, vol 165, issue 10. Water management, pp. 567–580
- Stelling GS, Kernkamp HWJ, Laguzzi MM (1998) Delft flooding system: a powerful tool for inundation assessment based upon a positive flow simulation. *Hydroinformatics* 449:456
- Sweers HE (1976) A nomogram to estimate the heat-exchange coefficient at the air-water interface as a function of wind speed and temperature: a critical survey of some literature. *J Hydrol* 30(375):401
- Umlauf L, Burchard H (2003) A generic length-scale equation for geophysical turbulence models. *J Mar Res* 61(31):235– 265
- Van't Hof B, Vollebregt EAH (2005) Modelling of wetting and drying of shallow water using artificial porosity. *Int J Numer Methods Fluids* 48(1199):1217

Autorenfassung

Sehili, Lang, Lippert: High-resolution subgrid models: background, grid generation, and implementation, 2014

Yu D, Lane SN (2006a) Urban fluvial flood modelling using a twodimensional diffusion-wave treatment, part 1: mesh resolution effects. Hydrol Process 20(7):1541–1565

Yu D, Lane SN (2006b) Urban fluvial flood modelling using a two-dimensional diffusion-wave treatment, part 2: development of a sub-grid-scale treatment. Hydrol Process 20(7):1567–1583