Response to the review of G. Munhoven

General comments

Appreciation of the manuscript
In this paper, E. V. Yakushev and co-authors present a highly complex model suitable to study the coupled biogeochemical processes at the bottom boundary layer, the sediment-water interface and the surface sediment. The model appears to provide an extremely complete description, considering all the processes and primary and secondary chemical reactions that have been taken into account. It is integrated into the *Framework for Aquatic Biogeochemical Models*, FABM (Bruggeman and Bolding, 2014).

Although the model appears to have been skilfully designed and set up, the paper has, unfortunately, a number of weaknesses. It is not suitable for publication in *Geosci- entific Model Development* in its current form – it should nevertheless be possible to **reconsider it after a major revision**.

This paper would definitely have benefited from another round or two of rereading and proofreading. Not even the name of the model is unambiguously given: in the title, the name is *Bottom RedOx Model*, in the model presentation (p. 2, ll. 2–3) it is *Bottom RedOx Layer Model*. The English of the paper needs some thorough revision. There are parts that are acceptable and others that are almost unsuitable for review. I am not going to point out all the **English** errors that I found – they are simply to numerous to key them all in here. There is one British co-author and two co-authors with affiliations to institutions in English-speaking countries or regions: could they please have a look at the manuscript and help to correct it and rewrite where necessary! There are errors (spelling, grammar, syntax, style) on nearly every single page, but sections 3.2.4 (Manganese) and 3.3 (Carbonate system) require particularly close attention.

The paper has been submitted as a “model description paper”. Requirements for that type of paper are detailed in http://www.geoscientific-model-development.net/about/ manuscript_types.html#item1. Quite some requirements are not met in this paper.

We ask the Reviewer to accept our sincere apologies for the poor condition of the submitted manuscript. We also wish to convey our gratitude to the Reviewer for nevertheless providing a very detailed and constructive review. We feel confident that this review, in combination with the other two, has contributed to a major improvement in the model code and description.

The model description is not well contextualized. The application presented is for a shallow-water environment, but one may ask where else it could be applicable, and which extensions or adaptations would be required or which simplifications would be possible. The authors mention, e. g., a possible coupling to NEMO,which encompasses almost the complete range of marine environments that one can imagine.

The Background section has been extended to improve the model contextualization. The broad applicability of BROM is now clarified in our final paragraph of the Background, where the distinguishing features of BROM vs. other models are listed:

"The goal of this work was to develop a model that captures key biogeochemical processes in the water and sediment and to analyze the changes occurring in the BBL and SWI. As a result, BROM differs from existing biogeochemical models in several key respects. BROM features explicit, detailed descriptions of many chemical transformations under different redox conditions, and tracks the fate of several chemical elements (Mn, Fe, S) and compounds ($MnCO_3$, FeS, S0, S_2O_3) that rarely appear in other models. BROM also allows for spatially explicit representations of the vertical structure in the sediments and BBL. This distinguishes it from e.g. ERSEM (Butenschon et al., 2016) which has a more detailed representation of benthic biology (meiofauna and different types of macrofauna), but limits its chemistry to the dissolved phase to CO_2, O_2 and macronutrients, and its vertical structure of sediments to an implicit three-layer representation that relies on equilibrium profiles of solutes and idealized profiles of particulates. Third, BROM offers a near-comprehensive representation of all processes affecting oxygen levels in the BBL and sediments, and should therefore provide a useful tool for studies focused on deoxygenation in deep water and sediments. Finally, BROM is conceived and programmed as a flexible model that can be applied in a broad range of marine and lake environments and modelling problems. As a component of the Framework for Aquatic Biogeochemical Modelling (FABM, Bruggeman and Bolding, 2014), BROM can be very easily coupled online to any hydrodynamic model within the FABM, and can also be driven offline by hydrodynamic model output saved in netCDF or ascii format (using the purpose-built offline transport solver BROM-transport)."

The technical details of the implementation are incomplete, and therefore, the criterion of model reproducibility that the paper should aim for is not met. All to many details are not covered in the description.

The level of technical detail in the new manuscript has been substantially increased. BROM-transport is now described in much greater detail in the text. The BROM-biogeochemistry description has been reworked and the Tables now provide an accurate and exhaustive description of all parameterizations and parameter values. We have added an Appendix guide "Running BROM step-by-step" and have made the input files (netCDF, fabm.yaml, and brom.yaml) for the demonstration run available on the BROM-transport git repository so that these results can be exactly reproduced. The .yaml input files that contain further technical details of implementation have also been provided as Appendices.

1. The instructions about where to get the code are incomplete. Much guesswork is currently required to locate the relevant files inside the FABM distribution. This could easily be avoided by, say, three to five extra sentences.

   This section has been improved and extended. We also added an Appendix “Running BROM step-by-step” to help the reader to run the model locally.

There does not seem to exist a way to permanently access the precise model version described in this paper.

Now we provide permanent tags for both BROM and FAMB: BROM-transport tag v1.1. https://github.com/e-yakushev/brom-git.git and the BROM-biogeochemistry code in FABM tag v0.95.3 http://fabm.net.

Also now there is a Win32 executable file available at https://github.com/e-yakushev/brom-git/releases/tag/v1.1

The limitations of the model and the fundamental software requirements are not given: if the model described here is really BROM-transport (this is not a name found in the paper, but it is the name of the only sensible source code collection that I could find), then the paper needs to state right away that:

- the BROM source code can only be compiled with the Intel Fortran compiler for Windows
- the current version can only use hydrodynamic conditions derived from GOTM (according to the Wiki at https://sourceforge.net/p/fabm/wiki/BROM_ FABM).

Although it is reported on p. 16 (l. 5), that the model was run with the Intel compiler for Windows,[1] it is said nowhere that this is the only way to run it. This is obviously a extremely strongly limitation and I am wondering whether such a restriction is

fundamentally necessary. As far as I can see, FABM itself is written in standard- conforming Fortran 2003 in a portable manner (no hardcoded kind types, etc.) and does not seem to rely on a single compiler for a single platform.

[1] It would be useful to provide the version number of the compiler used. FABM and BROM require some specific Fortran 2003 features and the Intel compiler only offers full support for Fortran 2003 since version 15. However, a subset offered by earlier versions might be sufficient here.
I strongly encourage the authors to prepare a version of the source code that can be used on other platforms and with alternative compilers. It should be possible to do this quite rapidly by introducing a few pre-processor directives, which would switch off some extra functionality provided by the Intel Fortran compiler for Windows, but that is not fundamentally required for the model itself. This would increase the usefulness of BROM by orders of magnitude! Else, what is the point in emphasizing that the model code "[. . .] uses modern software standards: it is coded in object-oriented Fortran 2003, […]" (p. 27, ll. 17–18) if in the end, it only compiles with one single compiler on one single platform.

The new version of BROM is platform independent and is currently used by the co-authors under both Windows and Linux.

The model itself seems to be carefully designed and set up. There are a few assumptions regarding the physical environment that may be debatable and that would benefit from a few extra words of explanation (see specific comments below). The set of processes and coupled chemical reactions and equilibria that have been taken into account is extremely complex. It is not obvious if such a high degree of complexity is truly necessary. The model indeed seems to allow a rather accurate simulation of the environment chosen. However, to what extent does it contribute to improve our understanding of the way the environment evolves? It would be interesting to know which are the dominant actors of the system. Unfortunately, the paper does not address this kind of question at all.

The physical environment assumed by the offline 1D solver BROM-transport is now described and explained in much greater detail. Regarding model complexity, we state the philosophy behind BROM in the new section 2.1.1:

"The model has 33 state variables (Ci), described in Table 1. This includes frequently measured components such as hydrogen sulfide (H2S) and phosphate (PO4), as well as rarely measured variables such as elemental sulfur (S0), thiosulfate (S2O3), trivalent manganese species Mn(III), and bacteria. Variables of the latter category were included because their contribution to biogeochemical transformations is believed to be substantial. For instance, bacteria play an important role in many modelled processes and can consume or release nutrients in organic and inorganic forms (Canfield et al., 2005; Kappler et al., 2005). We acknowledge that for many of these additional variables, site-specific estimates of associated model parameters and initial/boundary conditions may be difficult or impossible to obtain, and may in practice require some crude assumptions and approximations (e.g. universal default parameter values, no-flux boundary conditions, initial conditions from a steady annual cycle). Nevertheless, we believe that for many applications this will be a price worth paying for the additional process resolution/realism provided by BROM for important biogeochemical processes in the BBL and sediments."

Regarding the contribution to understanding through model analysis: This is a very important message but we believe that it requires a special study that is beyond the scope of the present description paper. However, we do plan to perform such analysis with a model carefully validated to a natural system, as part of a separate publication.

Specific comments

Introduction

The scope of the model, i. e., the bottom boundary layer (BBL), (also known as the benthic boundary layer, or are there differences between those two BBLs?) deserves to be presented in more detail. What is its typical thickness? What influences that thickness? How does it change throughout the global ocean? What are the typical gradients across the BBL? Please do not forget that *Geoscientific Model Development* has a broad lectureship.

The BBL is indeed a focus of BROM, but it is not the only one: BROM also focuses on the upper layers of the sediment. Also, we are anxious not to lengthen the paper too much through extended discussion or literature review. BROM offers a novel and applicable tool to study water column plus sediment biogeochemistry in an integrated way and with a focus on redox chemistry and deoxygenation. We want readers to be able to quickly assess whether or not BROM will be useful to them, and to have a detailed documentation of the model if they decide to use it. Finally, BROM is not a specialized BBL model; it is rather a "benthic-pelagic" coupled model for the water column and sediments. As far as BROM is concerned, the BBL is simply a thin layer of calm water separating the "pelagic" water column from the sediments. The treatment of the BBL in the current version of BROM is quite simple: the vertical diffusivity is either set to a (low) constant value (the simplest assumption) or it increases linearly from the SWI (roughly corresponding to the assumption of a log layer for current speed, Boudreau and Jorgensen, 2001; Holtappels and Lorke, 2011). With these considerations and the comments of all reviewers in mind we have included the following paragraph in the new Background:

"The BROM model described herein is a fully coupled benthic-pelagic model with a special focus on deoxygenation and redox biogeochemistry in the sediments and Benthic Boundary Layer (BBL). The BBL

is "the part of the marine environment that is directly influenced by the presence of the interface between the bed and its overlying water" (Dade et al., 2001). Physical scientists tend to prefer the term "bottom boundary layer", but this is largely synonymous with the BBL (Thorpe, 2005). Within BROM, the term BBL is used to refer to the lower parts of the fluid bottom boundary layer where bottom friction strongly inhibits current speed and vertical mixing, hence including the viscous and logarithmic sublayers up to at most a few metres above the sediment. This calm-water layer plays a critical role in mediating the interaction of the water column and sediment biogeochemistry and in determining e.g. near-bottom oxygen levels, yet it remains poorly resolved in most physical circulation models. For BROM we have developed an accompanying offline transport module "BROM-transport" that uses output from hydrodynamic water column models but solves the advection-diffusion-reaction equations for a "full" grid including both water column and sediments. BROM-transport uses greatly increased spatial resolution near to the SWI, and thereby provides explicit spatial resolution of the BBL and sediments."

I am surprised to read that “the Bottom Boundary Layer (BBL) […] is still understudied” (p. 5, l. 22). On my shelf I have the fine book *The Benthic Boundary Layer: Trans- port Processes and Biogeochemistry* (Boudreau and Jørgensen, 2001). It is nearly 15 years old and BBL research has certainly not come to a rest since that book got published. Please reconsider that statement and provide a fair representation of the existing literature.

We have removed this statement. The summary of existing literature in the Background section has also been expanded to provide a fairer representation.

General model presentation

Scope of the model

In the end, it is not entirely clear what the exact scope of BROM is. In the abstract, BROM is introduced as a model for the biogeochemical process in the bottom boundary layer; in the model description though, we read that “[t]he water column considered in our model spans the sea surface (upper boundary) down to user’s defined sediment depth [. . .]”. This is to some extent contradicting as this domain largely exceeds the bottom boundary layer. Please clarify.

BROM was never intended to cover only the bottom boundary layer or to exclusively focus on this. We apologize for the lack of clarity in the original submission. In the new manuscript we have harmonized and clarified the stated goals and scope. In the new Abstract we have:

"The goal of this work was to develop a model that captures key biogeochemical processes in the water and sediments and that simulates the changes occurring in the bottom boundary layer and sediment-water interface."

then in the new Background section we have:

"The goal of this work was to develop a model that captures key biogeochemical processes in the water and sediment and to analyze the changes occurring in the BBL and SWI."

Computational aspects

It is stated that numerical integration was carried out with the Eulerian scheme (the explicit or the implicit variant? – the extremely short time-steps chosen make me guess it is the former, but it would be good to state this). Is the same Eulerian scheme used for both space and time dimensions? Please specify all the schemes used.

The numerical integration options of BROM transport are now described in a dedicated section 2.2.2. The new text reads:

"Equations (1-3) are integrated numerically over a single combined grid (water column plus sediments) and using the same model time step in both water column and sediments. All concentrations are stored internally and input/output in units [mmol/m^3 total volume]. Time stepping follows an operator splitting approach (Butenschon et al., 2012): concentrations are successively updated by contributions over one time step of diffusion, bioirrigation, reaction, and advection, in that order. If any state variable has any 'not-a-number' values at the end of the time step then the program is terminated.

Diffusive updates are calculated either by a simple forward-time central-space (FTCS) algorithm or by a semi-implicit, central-space algorithm adapted from a routine in the General Ocean Turbulence Model (GOTM, Umlauf et al., 2005). Bioirrigation and reaction updates are calculated as forward Euler time steps, using the FABM to compute R_i, and advection updates are calculated using a simple first-order upwind differencing scheme. After each update, Dirichlet boundary conditions (see below) are reimposed and all concentrations are low-bounded by a minimum value (default = 10^{-11} µM) to avoid negative values.

BROM-transport also provides the ability to divide the diffusion and advection updates into smaller time steps related to the sources-minus-sinks time step by fixed factors, since the physical transport processes are often numerically limiting (Butenschon et al., 2012). The default time step is 0.0025 days or 216 seconds, which is much longer than the characteristic equilibration timescale of the CO_2 kinetics (Zeebe and Wolf-Gladrow, 2001)."

Details about the pH solving algorithm can only be looked up in the code. The text only says that "[…] total pH was calculated using the Newton-Raphson method" (p. 11, ll. 20–21) and that "Carbonate system equilibration was parameterized using the standard approach (i.e. Lewis and Wallace, 1998)" (Table 2). This latter affirmation is meaningless: Lewis and Wallace (1998) neither provide information about the methods used in their program, nor do they define any standard approach. A few more details about how calculations are actually done in BROM would be of order here.

The carbonate system code was updated, in particular we added dependence of the carbonic acid constants on pressure, and we implemented the pH calculation method proposed by Munhoven (2013).

In the new text, the methods for calculating the carbonate system are described in section 2.1.4:

"Equilibration of the carbonate system was considered as a fast process occurring within a few seconds (Zeebe and Wolf-Gladrow, 2001). Accordingly, the equilibrium solution was calculated at every time step using an iterative procedure. The carbonate system was described using standard approaches

(Munhoven, 2013; Roy et al., 1993; Wanninkhof, 2014; Wolf-Gladrow et al., 2007; Zeebe and Wolf-Gladrow, 2001). The set of constants of (Roy et al., 1993) was used for carbonic acid. Constants for boric, hydrofluoric, and hydrogen sulfate alkalinity were calculated according to (Dickson, 1992), for silicic alkalinity according to (Millero, 1995), for ammonia alkalinity according to (Luff et al., 2001), and for hydrogen sulfide alkalinity according to (Luff et al., 2001) and (Volkov, 1984). The ion product of water was calculated according to (Millero, 1995). Total scale pH was calculated using the Newton-Raphson method with the modifications proposed in (Munhoven, 2013). Precipitation and dissolution of calcium carbonate were modelled following the approach of (Luff et al., 2001) (Table 2)."

In general, the text really ought to be more complete and informative about **numerical aspects of the model**. This is what *Geoscientific Model Development* readers expect.

We have addressed numerical aspects more thoroughly in the new text, including dedicated sections on numerical integration and numerical details on the carbonate system calculation (see above).

Rate law expressions

The tables that list all the processes considered in BROM and their rate laws, and that collect the different parameter values are among the most informative parts of the paper. They clearly represent one of its major strengths. Unfortunately no references are given for the parameter values presented in Table 3. There is a large variety of kinetic rate laws that are used in the model (Monod laws, squared Monod laws, laws in tanh, . . .). I think it would be good to have a few words of explanation about the choices operated. Please also complete the references where missing (Table 2, on pp. 41–43 and Table 3, throughout).

We have checked and completed the references in Table 2 and added references for the coefficient values in Table 3. Regarding the use of squared availabilities, an explanation has been added to the text:

“The redox-dependent switches are preferably based on hyperbolic functions that improve system stability compared with discrete switches. The nutrient limitation and trophic functions are preferably based on squared Monod laws for Nutrient/Biomass ratio, which also stabilizes the system compared with Michaelis-Menten and Ivlev formulations.”

As mentioned in the general appreciation already, I really wonder if all that complexity is really necessary, or, put the other way around: which minimalist set of process would be sufficient to obtain realistic results?

Motivation for the complexity of BROM is provided in the Background section of the new text:

“The goal of this work was to develop a model that captures key biogeochemical processes in the water and sediment and to analyze the changes occurring in the BBL and SWI. As a result, BROM differs from existing biogeochemical models in several key respects. BROM features explicit, detailed descriptions of many chemical transformations under different redox conditions, and tracks the fate of several chemical

elements (Mn, Fe, and S) and compounds ($MnCO_3$, FeS, S0, S_2O_3) that rarely appear in other models. BROM also allows for spatially explicit representations of the vertical structure in the sediments and BBL. This distinguishes it from e.g. ERSEM (Butenschön et al., 2015), which has a more detailed representation of benthic biology (meiofauna and different types of macrofauna), but limits its chemistry to the dissolved phase to CO_2, O_2 and macronutrients, and its vertical structure of sediments to an implicit three-layer representation that relies on equilibrium profiles of solutes and idealized profiles of particulates. Third, BROM offers a near-comprehensive representation of all processes affecting oxygen levels in the BBL and sediments, and should therefore provide a useful tool for studies focused on deoxygenation in deep water and sediments. "

Further explanation of the BROM philosophy regarding model complexity has also been added to section 2.1.1:

"The model has 33 state variables (Ci), described in Table 1. This includes frequently measured components such as hydrogen sulfide (H_2S) and phosphate (PO_4), as well as rarely measured variables such as elemental sulfur (S0), thiosulfate (S_2O_3), trivalent manganese species Mn(III), and bacteria. Variables of the latter category were included because their contribution to biogeochemical transformations is believed to be substantial. For instance, bacteria play an important role in many modelled processes and can consume or release nutrients in organic and inorganic forms (Canfield et al., 2005; Kappler et al., 2005). We acknowledge that for many of these additional variables, site-specific estimates of associated model parameters and initial/boundary conditions may be difficult or impossible to obtain, and may in practice require some crude assumptions and approximations (e.g. universal default parameter values, no-flux boundary conditions, initial conditions from a steady annual cycle). Nevertheless, we believe that for many applications this will be a price worth paying for the additional process resolution/realism provided by BROM for important biogeochemical processes in the BBL and sediments."

Miscellanea

Denitrification is considered, and nitrification, but I could not find anything about how nitrogen fixation is dealt with. I would expect that this process is required to avoid an unrealistic drift in the nitrogen inventory.

Corrected.

Total alkalinity

This part of the paper (p. 9) is one of the most disappointing ones. It is very approxi- mate, completely overloaded with information that is ignored in the end. It furthermore contains several errors.

For clarity, it would be best to provide immediately the approximation actually used in the model, and not a hypothetical one, that could have been used. Alkalinity contribu- tions that are not included or that are set to zero should be omitted. The text will be considerably simplified.

Whatever the expression chosen for total alkalinity, it will anyway always remain only an approximation. But even approximations need to be factually correct. Unlike written in the paper,.

- ... H_3PO_4 is also part of alkalinity and $A_{TPO_4} = [HPO_4] + 2[PO_4] - [H_3PO_4]$ — interestingly this is correct in Table 6 (except for a typo) and also in the code;
- ... NH_4^+ is not part of alkalinity (it is the zero-level species) and thus $A_{TNH3} = [NH_3]$;

- . . . it is the total borate concentration that is estimated from salinity and not $[B(OH)_4^-]$ — $[B(OH)_4^-]$ is calculated from the state variables just like to others (this is correctly done in the code, fortunately);
- . . . F^- is not part of alkalinity, only HF, so that A_{THF} = [HF] — this is also wrong in Table 6 (at 68 μM, it would be barely negligible), but I suggest to discard the A_{THF} term from the alkalinity expression anyway, as it is not included in the model.
-

We apologize for these errors and lack of clarity in the submitted text. The total alkalinity formulation has now been corrected. We have chosen to retain the more general expression for total alkalinity as a starting point and then explicitly neglect the hydrogen sulfate, hydroflouric and nitrous acid terms. We feel that this helps to link our approach with the "classical" formulation.

Although it is specified later on that the stoichiometric constants of Roy et al. (1993) are used for the carbonate system, references for the other constants (e. g., dissociation constants for boric, phosphoric and silicic acids) required to solve the total alkalinity-pH equation are missing. Please provide references for those as well.

The references are now provided in the code (fabm.yaml) and the text:

“The set of constants of (Roy et al., 1993) was used for carbonic acid. Constants for boric, hydrofluoric, and hydrogen sulfate alkalinity were calculated according to (Dickson, 1992), for silicic alkalinity according to (Millero, 1995), for ammonia alkalinity according to (Luff et al., 2001), and for hydrogen sulfide alkalinity according to (Luff et al., 2001) and (Volkov, 1984). The ion product of water was calculated according to (Millero, 1995)."

Finally, the pH scale used in the paper turns out to be the total scale. This should be stated more clearly than it is currently done (at my third reading, I discovered on p. 11 (l. 20) that “total pH was calculated”. Please state this more obviously.

We now specify total scale pH twice when the variable is first mentioned in section 2.1:

"Instead, the total scale pH is calculated as a diagnostic variable at every time step as a function of DIC and Alk (which are state variables). In turn, the total scale pH is used in calculations of the chemical equilibrium constants required to describe related processes (i.e. carbonate precipitation/dissolution, carbonate system parameters etc.)."

Physical environment

Porosity

Variable porosity is not included in the current version of BROM. The affirmation that “[. . .] its effect on [the] vertical transport is incorporated in[to] the values of K_z and $K_{z\text{bio}}$, […]” (p. 13, l. 17) is rather obscure. $K_{z\text{mol}}$ is actually constant so it is not clear how it could take porosity variations into account. I am furthermore not certain that this simplification is really necessary, given the complexity and detailed representation of the rest of the model. Variable porosity should not significantly increase the model’s compleity.Furthermore, it appears that a tortuosity corresponding to the porosity value of 90%

was used, with reference to a “value from Boudreau, 1997” (p. 13, l. 22). This is not very meaningful. Boudreau (1997) lists eight theoretically based tortuosity-porosity relationships and three empirical ones. Please specify which one was used here and then cite the original reference.

We agree: this simplification was excessive. BROM-transport has received a major overhaul and now includes variable porosity as a fixed profile following the parameterization of Soetaert et al. (1996). Porosity now distinguishes the solute from particulate dynamics within the sediments assuming intraphase mixing (Boudreau 1997; see section 2.2.1); its effects are now treated explicitly and not folded into the vertical diffusivity (which in fact cannot fully account for porosity variations). The apparent or effective molecular diffusivity now varies with depth due to variable tortuosity. This is described in the new section 2.2.1:

"The total solute diffusivity $D_C = D_m + D_B$, where D_m is the apparent molecular/ionic diffusivity and D_B is the bioturbation diffusivity due to animal movement and ingestion/excretion. The apparent molecular diffusivity $D_m(z) = \theta^{-2} D_0 \frac{\mu_0}{\mu_{sw}}$ is derived from the infinite-dilution molecular diffusivity D_0 (an input parameter) assuming a constant relative dynamic viscosity $\frac{\mu_0}{\mu_{sw}}$ (default value 0.94, cf. Boudreau 1997, Table 4.10) and a tortuosity parameterized as: $\theta^2 = 1 - 2\ln\varphi$ from Boudreau (1997) Eqn. 4.120."

Boudreau (1997) is actually the original reference for this tortuosity parameterization. Boudreau himself refers to it as a "modified Weissberg" relation (Boudreau, 1997; Eqn. 4.119) but the empirical fit of the constant "b" is due to Boudreau (Boudreau, 1997; Table 4.12, Fig. 4.10).

Molecular diffusion

BROM uses a species-independent molecular diffusion coefficient. This consider- ably simplifies the advection-diffusion-reaction equations, as the total concentrations a, such as DIC and alkalinity can be transported directly. The reported value $K_{z\mathrm{mol}} = 1 \times 10^{-11}$ m^{2}s^{-1} is, however, almost two orders of magnitude lower than those for typ- ical ions: e. g., from Boudreau (1997, Table 4.8), we may calculate diffusion coeffi- cient values of 0.781×10^{-9} m^{2}s^{-1} for HCO^-_3, 0.632×10^{-9} m^{2}s^{-1} for CO^{2-}_3 and even 1.313×10^{-9} m^{2}s^{-1} for HS^- (each one for $t = 10\,^\circ$C). These are infinite dilution diffusion coefficients, but correcting them for tortuosity and for the dynamic viscosity of seawa- ter does not reduce these values by more than 15–20%. How would results change if these much higher values would be used?

We agree that the species-independent molecular diffusivity is a simplification, but as the reviewer states it does substantially simply matters in regard to composite variables. We have retained the species independence in the new code, although as it is now written the user would only have to make small modification to the code to allow species dependent diffusivity ($K_{z\mathrm{mol}}$ in the new code is actually stored as a matrix over depth and state variable, with zeros for particulate variables). We agree that the previous default value was too small, even if assumed to account for tortuosity and dynamic viscosity. In the new version, the default (single) value for infinite-dilution molecular diffusivity is 1×10^{-9} m^{2}s^{-1} based on the coefficients in Boudreau (1997, Table 4.8) (see brom.yaml in Appendix D). The user is free to change this parameter in the run-time brom.yaml file, where we also state the default value as well as a plausible range $(0.5\text{-}2.7) \times 10^{-9}$ m^{2}s^{-1} again derived from Boudreau (1997, Table 4.8). The default value of 1×10^{-9} m^{2}s^{-1} was used for the demonstration simulation in the new section 3 (see Appendix D).

Bioturbation

Biotubation is parametrized as a diffusive process, as is common usage. For the biod- iffusion coefficient, it is only stated that it takes a constant value over the top 2 cm and that it decreases exponentially afterwards. However, I have not been able to find the length scale of this decrease anywhere in the text. Now, one may ask whether it is re- alistic to consider any bioturbation at all in anoxic parts of the sediment, the more since the text already indicates that the maximal bioturbation depth was only 0.5–2.2 cm (p. 13, ll. 15–16). How would this change your conclusions?

We apologize for making the reviewer search to no avail. The exponential decay scale is a user-defined parameter defined in the brom.yaml file (see Appendix D). Here we specify a default value of 1 cm, citing Soetaert and Middelburg (2009). In anoxic conditions, the entire profile of bioturbation diffusivity is scaled down by a Michaelis Menten function of oxygen concentration at the sediment surface. This is described in the new section 2.2.1:

"The bioturbation diffusivity $D_B(z,t)$ is modelled as a Michaelis-Menten function of the dissolved oxygen concentration in the bottom layer of the water column:

$$D_B(z,t) = D_{Bmax}(z)\frac{O_{2s}}{O_{2s}+K_{O2s}} \qquad (5)$$

where $D_{Bmax}(z)$ is a constant over a fixed mixed layer depth in the surface sediments then decays to zero with increasing depth, and K_{O2s} is a half-saturation constant. The rationale for (5) is that the animals (worms etc.) that cause bioturbation require a source of oxygen at the sediment surface for respiration."

Bioirrigation

BROM takes the important process of bioirrigation into account. It is, however, represented as a purely diffusive process. Boudreau (1997) and Aller (2001) make a strong case that it would be more appropriate to represent bioirrigation as a non-local exchange process instead.

The simple possible parameterization, probably it is not enough… Of cource there are other approaches, but …

We agree. A more thorough examination of the literature shows little theoretical or observational support for a local diffusive parameterization of bioirrigation. In the new BROM-transport model, bioirrigation is modelled as a non-local exchange process as proposed in Boudreau (1997). This is described in the new section 2.2.1:

"Finally, the process of bioirrigation, whereby worms flush out their burrows with water from the sediment surface, is modelled as a non-local solute exchange following (Meile et al., 2001; Rutgers Van Der Loeff and Boudreau, 1997; Schlüter et al., 2000):

$$T_{birrC(i)} = \alpha\varphi\frac{O_{2s}}{O_{2s}+K_{O2s}}\left(\hat{C}_{f(i)} - C_i\right) \quad \text{(for solutes)} \qquad (10)$$

where $\alpha(z)$ is the bioirrigation rate in oxic conditions, $\hat{C}_{f(i)}$ is the flushing concentration of solute in the fluff layer, and the Michaelis-Menten function again accounts for the suppression of worm activity in anoxic conditions. The oxic bioirrigation rate $\alpha(z)$ is parameterized as an exponential decay from the sediment surface as in Schluter et al. (2000). The total mass transfer to/from the sediment column must be balanced by a flux into/out of the fluff layer (see equation (1)):

$$T_{birr(i)} = \frac{1}{h_f}\frac{O_{2s}}{O_{2s}+K_{O2s}}\int_{z_{SWI}}^{z_{max}} \alpha\varphi\left(C_i - \hat{C}_{f(i)}\right) dz' \qquad \text{(for solutes)} \qquad (11)$$

where h_f is the thickness of the fluff layer and z_{max} is the depth of the bottom of the modelled sediment column. $T_{birrC(i)}, T_{birr(i)} = 0$ for all particulate variables."

Code

On p. 7 (ll. 24–25), it is said that BROM consists of three modules. I did not want to download and install the complete FABM, but nevertheless wanted to inspect the BROM code, to find out more about the technical details that were missing from the paper. This was, however, not entirely straightforward.

Accessibility

After having opened http://fabm.org (which redirects to the FABM project page on SourceForge), I started to search for references to BROM. After some searching around, I detected the first trace of BROM under the “Wiki” tab: section 7 of chap- ter 2 of the User’s Guide has the title “BROM-transport + FABM”. BROM-transport is most probably the transport model mentioned in the paper (p. 7, l. 2), but that is not clear, since the paper always mentions BROM only. That section provides at least the first useful hint about where to find the BROM biogeochemical modules: under src/models/niva/brom in FABM. Proceeding to the “Code” tab then allowed me to browse to the relevant files (under the indicated directory tree). BROM-transport, however, is not with FABM and must be retrieved from a different repository, located at https://github.com/e-yakushev/BROM-transport, not mentioned in the paper. I suggest that the authors give accurate and comprehensive instructions in the paper

about where the actual BROM source code files are located, both the biogeochemical ones and the main driver. And, please include also information about the license under which the code is distributed.

We apologize for the confusion and wild goose chase. In the new text we clarified and extended the section "Code Availability". It now reads:

"The model as presented consists of two components. The first is a set of biogeochemical modules (brom/redox, brom/bio, brom/carb, brom/eqconst), available as part of the official FABM distribution (http://fabm.net) (for a currently-functional direct link please see https://sourceforge.net/p/fabm/code/ci/master/tree/src/models/niva/brom/). The second is a hydrophysical driver (BROM-transport) that provides the 1D vertical context and resolves transport; this is available separately from https://github.com/e-yakushev/brom-git.git. When combined, the 1D BROM model as presented is obtained. Additionally, as BROM’s biogeochemical modules are built on FABM, they can be used from a wide range of 1D and 3D hydrodynamic models, including GOTM, GETM, MOM, NEMO and FVCOM (NEMO-FABM and FVCOM-FABM couplers have been developed by the Plymouth Marine Laboratory; contact J.B. for information).

BROM biogeochemical modules follow FABM conventions: they are coded in object-oriented Fortran 2003, have a build system based on CMake, and use YAML files for run-time configuration. The code is platform independent and only requires a Fortran-2003-capable compiler, e.g., gfortran 4.7 or higher, or the Intel Fortran compiler version 12.1 or higher. The BROM-specific source code is located in the FABM code tree in directory src/models/niva/brom. The specific version used to produce the results described in this paper is associated with git commit 1581186939a0ff81a230468694bf909a42afc21e. However, we envisage the model to be further developed in a backward compatible manner, and encourage users to use the latest code version.

BROM-transport is coded in Fortran 2003. It includes facilities for producing results as NetCDF files, which can be read by a variety of software on different platforms. The reader should be able to reproduce the results shown in this paper using the BROM-transport and BROM-biogeochemistry code from the above repositories and the netCDF/.yaml input files found in the data/ folder of the BROM-transport repository. Step-by-step instructions for running BROM are found in Appendix A. BROM-transport as well as BROM biogeochemical modules are distributed under the GNU General Public License (http://www.gnu.org/licenses/)."

Code quality

The code is obviously “work in progress” and appears to undergo continuous changes. There are many lines of code that are commented out, some of them might be important. It is of not clear if they were also commented out when the results described in the paper were calculated.

The code quality and presentation have undergone a major overhaul. We have uploaded a finished, stable version with all commented-out code deleted. The reader should be able to reproduce the results shown in this paper using the BROM-transport and BROM-biogeochemistry code from the above repositories and the netCDF/.yaml input files found in the data/ folder of the BROM-transport repository (these .yaml files are also shown in Appendices C and D). Step-by-step instructions for running BROM are found in Appendix A.

I detected a few coding choices that put portability at risk. While REALs in the three biogeochemistry related modules are declared in a portable way with REAL(rk), where rk is an INTEGER parameter whose value gets derived from an appropriate SELECTED_REAL_KIND(...) call, there are some INTEGER(4) declarations that may lead to problems. In BROM-transport, there are numerous REAL(8) declarations, in different source code files. Kind type values – such as the ‘4’ of the INTEGER(4) or the ‘8’ of the REAL(8)declarations – are not standardized and may differ from one compiler to another. Programmers may not assume that they are equal to the ex- pected byte length and for portability reasons kind type values must therefore not be hard-coded.[2] Portable and reliable code would consistently follow the FABM approach, with the rkparameter derived from SELECTED_REAL_KIND(...)

All REAL(8)declarations in BROM-transport have now been changed to REAL(rk)where rk is inherited from the SELECTED_REAL_KIND(...)statement in the FABM code, using a command: use fabm_types, only: rk. All INTEGER(4)declarations have been replaced with INTEGER.

I have come across a few peculiarities or short-cuts in the code that may lead to seri- ous confusion: e. g., in the subroutine phIter in brom_carb.F90, the INTENT(IN) argument Sit_ (the total silicate concentration) is overridden by a local variable Sit,

[2] I know of one compiler where DOUBLE PRECISION is not REAL(8) but REAL(3).

which is set to zero, thus making the code ignore silicate alkalinity. The paper does, however, not state that silicate alkalinity is ignored.

The code has been significantly modified and the mentioned peculiarities have been removed.

The pH calculation routine is neither safeguarded nor does it include diagnostics for possible convergence failures or for early convergence: it simply executes 100 Newton- Raphson iterations, starting from a preset fixed starting value, that furthermore seems to require manual modification from time to time. No diagnostic is included, neither for possible convergence failures nor for early convergence. (Why carry out 100 iterations if convergence is reached after five of them already?)

There are now reliable methods to solve the alkalinity-pH equation, which are guaranteed to converge under any physically meaningful conditions, howsoever exotic, and usually in less than six iterations (**Munhoven**, 2013). These would be particularly recommended in the environments that BROM has been developed for, with complex alkalinity compositions and unusual total concentrations.

We are very grateful to the Reviewer for this suggestion, and have implemented the recommended methods to solve the alkalinity-pH equation. This has been very helpful in regard to computational efficiency.

Carbonate solubility constants do not take any pressure correction into account (the relevant lines are present, but commented out).

Corrected.

Finally, the comments in the code are not always correct, which also creates unnecessary confusion (e. g., the phosphoric alkalinity is not [H2PO4-] + 2.*[HPO4--] + 3.*[PO4-

--] as stated in a comment, but [HPO4--] + 2.*[PO4---] - [H3PO4]. Fortunately it is the latter that is implemented in the code.

Corrected.

Permanent access to the code for model version 1.0

As mentioned in the general appreciation, for model description papers there should exist a way to permanently access the precise model version described in the paper. The GitHub repository for BROM-transport includes a Ver. 1.0 directory, so for the transport model, this seems to be conceivable.

The biogeochemical modules that are hosted in the FABM repository are however not clearly tied to version 1.0 of BROM.

It would thus be necessary to provide somehow tagged versions of the source code files for the model version 1.0 described here, or to provide copies of those files as a supplement to the paper.

The tag 1.1 for BROM transport is provided, https://github.com/e-yakushev/brom-git/releases/tag/v1.1

2.6 Tables

The tables contain a wealth of information and represent one of the most useful parts of the paper (with the exception of Table 6, which could be deleted without loss). Un- fortunately, Tables 1 and 4 are nearly unreadable because of the small font size. They would clearly benefit from a reorganization of their contents. Table 2 currently spans eight pages, Table 3 six pages. It would be useful to split them into smaller parts, with dedicated captions. While Table 2 still contains extensive references, Table 3 does not contain a single one. Readers ought to know where the adopted parameter values come from or how they have been derived.

The tables have been modified following the Reviewer's suggestions.

The second column of the row "Alkalinity changes" in Table 2 is completely overloaded. Please reorganize this information.

Corrected.

Table 6 is not essential for the paper and I suggest to delete it altogether. It also contains errors and except for Canfield et al. (2005), none of the references cited is in the reference list. A_{THF} is certainly not 68 μM, else it would not be negligible.

Table 6 has been deleted.

3 Technical comments

Throughout the paper: change "protolithic" to "protolitic" or "equilibrium" (depending on the context)

Corrected.

Throughout the paper: change "connected with" to "related to"

Corrected.

Throughout the paper: please check the usage of the word “parameterized” and “pa- rameterization”. For example, in Table 2, it is said that the carbonate system equilibra- tion was parameterized. It were rather the stoichiometric constants that were parame- terized, as a function of temperature, salinity and pressure, but the carbonate system equilibration (it would be more correct to say speciation) was calculated.

Corrected.

p. 4, l. 26: “death or flight”? “death or migration” would perhaps be more appropriate

Corrected.

p. 7, l. 15: “changeable” is not appropriate in this context. Perhaps “varying”?

Corrected.

p. 9, ll. 20–25: it is common usage to speak about bor*ate*, phosph*ate* and silic*ate* alkalinity (as with *carbonate alkalinity*) and to reserve the terms bor*ic*, phosphor*ic* and silic*ic* for the corresponding acids (as in *carbonic acid*).

Corrected.

p. 11, l. 20: change “Roy’s constants” to “the set of constants of Roy et al.” – the co-authors will appreciate

Corrected.

p. 16, ll. 4–5: change “FORTRAN” to “Fortran 2003” (spelling and standard) and change “Intel FORTRAN for Windows Compiler” to “Intel Fortran Compiler for Win- dows”, which is the name of the product.

Corrected.

p. 16, l. 6: what is meant by “balanced distribution”?

We meant balanced fluxes in a quasi-stationary sense. This term has been deleted.

pp. 21–26 (section 3.2.4 – section 3.4): please check for the English and rewrite where necessary.

This has been done.

p. 39, rows 10 and 11: “sulfatereduction” should read “sulfate reduction”

Corrected.

p. 40, second-last row, right-hand column: should the "CaCO3" on the last linee not read "caco3_diss−caco3_prec"?

Corrected. Note that in the new Table 2.3, "caco3_prec" has been replaced with "caco3_form".

p. 41, row 7: there is probably some "NO3"-"NH3" mismatch here

Corrected. The correct equation reads:

$$\mathrm{LimNO_3} = \frac{((\mathbf{NO_3}+\mathbf{NO_2})/\mathbf{Phy})^2}{\mathrm{K\_nox\_lim}^2+((\mathbf{NO_3}+\mathbf{NO_2})/\mathbf{Phy})^2}\exp(-\mathrm{K\_psi}\frac{(\mathbf{NH_4}/\mathbf{Phy})^2}{\mathrm{K\_nh4\_lim}^2+(\mathbf{NH_4}/\mathbf{Phy})^2})$$

p. 41, rows 7 and 8: the two trailing '2's in exponent seem to be misplaced (they probably belong to the second term in the denominator each time)

Corrected. Please forgive our sloppy editing.

p. 46, in the first row relative to a half-saturation for OM denitrification, "NO2" should probably read "NO3"

Corrected.

Table 6: "[PO^{2-}_4]" should read "[PO^{3-}_4]"

This table has been deleted.

References

Aller, R. C.: Transport and reactions in the bioirrigated zone, in: The Benthic Boundary Layer : Transport Processes and Biogeochemistry, edited by Boudreau, B. P. and Jørgensen, B. B., chap. 11, pp. 269–301, Oxford University Press, New York (NY), 2001.

Boudreau, B. P.: Diagenetic Models and Their Implementation, Springer-Verlag, Berlin, 1997.

Boudreau, B. P. and Jørgensen, B. B., eds.: The Benthic Boundary Layer : Transport Processes and Biogeochemistry, Oxford University Press, New York (NY), 2001.

Bruggeman, J. and Bolding, K.: A general framework for aquatic biogeochemical models, Env- iron. Model. Softw., 61, 249–265, doi:10.1016/j.envsoft.2014.04.002, 2014.

Lewis, E. and Wallace, D.: Program developed for CO2 system calculations, Tech. Rep. 105, Carbon Dioxide Analysis Center, Oak Ridge National Laboratory, Oak Ridge (TN), available at http://cdiac.ornl.gov/oceans/co2rprt.html, 1998.

Munhoven, G.: Mathematics of the total alkalinity-pH equation – pathway to robust and uni- versal solution algorithms: the SolveSAPHE package v1.0.1, Geosci. Model Dev., 6, 1367– 1388, doi:10.5194/gmd-6-1367-2013, 2013.

Roy, R. N., Roy, L. N., Vogel, K. M., Porter-Moore, C., Pearson, T., Good, C. E., Millero, F. J., and Campbell, D. M.: The dissociation constants of carbonic acid in seawater at salinities 5 to 45 and temperatures 0 to 45 °C, Mar. Chem., 44, 249–267, doi:10.1016/0304-4203(93)90207-5, 1993.