r.avaflow v1, an advanced open source computational frame-work for the propagation and interaction of two-phase mass flows

Martin Mergili, Jan-Thomas Fischer, Julia Krenn and Shiva P. Pudasaini

Response to the comments of Referee #1

We would like to thank the reviewer for her or his constructive remarks. Below, we address each comment in full detail. We have considered all suggestions when revising the manuscript. Changes in the manuscript are highlighted in yellow colour.

The paper presents a complex dynamic model for simulation of interacting solid and fluid mixture flows implemented in open source GRASS GIS. It represents an innovative, valuable contribution to the field of mass flows modeling. As authors point out in the discussion, the model requires number of parameters which are not readily available and validation of the model for real-world situation will require a complex experimental set up. Therefore, the model now serves mostly as a valuable theoretical tool to improve understanding of the complex mass flow processes. Implementation in a widely used open source GIS will provide opportunities for collaborations in validation and improvement of the model.

The paper is well written, although in general it is easier to read papers which start with theory and mathematical foundations of the model followed by the implementation but in this case the reverse structure is acceptable. The core of the model is apparently described by Pudasaini (2012) but it would be useful to accurately specify the topography-following coordinates, as the description on p. 8 l. 220-224 is rather vague.

Thank you for this comment, you are pointing out a very important issue here, which in the revised manuscript is dealt with in lines *83–86* (different coordinate systems), *232–240* (model equations – numerics – GIS) and finally in lines *594–601* (description of the solution with complementary function). To avoid ambiguity we reformulated these parts as follows:

Lines 83–86: This issue is closely related to the fact that the model equations are commonly expressed in topography-following coordinates hardly compatible with global Cartesian coordinates, which usually appear in Geographic Information Systems (GIS) and are referred to as GIS coordinates in the following.

Lines 232–240: The input and output of r.avaflow (see Sect. 2.2) is discretized on the basis of GIS coordinates, i.e. in cells which are rectangular in shape in the ground projection. For the numerical solution the cell lengths in x and y directions, and the area, are corrected for the local slope in order to maintain consistency with the state variables expressed in the local topography-following coordinates. Gravitational acceleration in the topography- following x, y, and z directions – representing a fundamental input to the Pudasaini (2012) model equations – is computed from the DTM, employing a finite central difference scheme. All input heights H (m) are expressed in vertical direction, and are converted into depths D (m) expressed in direction normal to the local topography as in the model equation formulation. The resulting depths are converted into heights for output.

Lines 594–601: r.avaflow represents a modular framework, allowing for the future enhancement of its particular components. One issue concerns the numerical implementation of the two-phase model equations, combining topography-following *coordinates with the quadratic cells of the raster data given in GIS coordinates (see Sect. 2.3). As in comparable simulation tools (e.g. Christen et al., 2010a, b; Hergarten and Robl, 2015), approximations are currently used for coordinate transformation in r.avaflow. This issue is closely related to the fact that the model equations that are commonly expressed in topography-following coordinates are hardly compatible with the data given in GIS coordinates.*

A small formal issue- in the Figure 12 the relation between the colors in the image and the legends is not clear - it either needs to be explained in the caption (e.g., where is the blue and purple in the image?) If the legend is designed for the animation it would be better to have a different legend for the static image to make the image easier to interpret.

The legend is standardized in order toallow a straightforward comparison of case studies – or of various phases of one case study – with different solid contents (range of 0.0–1.0). As the Acheron rock avalanche mostly consists of solid, the colour is rather brownish, in contrast to the generic example where a broader range of fluid content is covered. This is valid for both the static images and the animations. This issue is clarified in the caption of Fig. 12 in the revised manuscript.

Just a final note that given that the module produces time series of raster maps – the series can be registered with GRASS GIS temporal framework and visualize easily as a dynamic surface in addition to the standard 2D animation.

We thank the reviewer for this suggestion.Yes, we are aware about the new space-time datatypes introduced with GRASS7. These possibilities have not been included in r.avaflow yet, but are most certainly a promising option for the future development of the tool.

r.avaflow v1, an advanced open source computational frame-work for the propagation and interaction of two-phase mass flows

Martin Mergili, Jan-Thomas Fischer, Julia Krenn and Shiva P. Pudasaini

Response to the comments of Julia Kowalski

We would like to thank the reviewer for her constructive remarks. Below, we address each comment in full detail. We have considered all suggestions when revising the manuscript. Changes in the manuscript are highlighted in yellow colour.

Review

The manuscript r.avaflow v1, an advanced open source computational framework for the propagation and interaction of two-phase mass flows describes a GRASS integrated software framework for the simulation of gravity-driven mass movements. Its underlying mathematical model and corresponding numerical solution are content of the author’s previous work. In this manuscript they focus on details regarding the integration of the numerical solution algorithm with a GRASS GIS environment, as well as additional functionalities that are needed when wanting to validate the simulation model with data. Performance and software framework are demonstrated during two test cases, one being different scenarios of a complex landslide in a synthesized topography, the other being the re-analysis of a New Zealand rock avalanche. The manuscript concludes with a discussion of necessary next steps and an outlook.

General impression

Generally, the paper is well written and easily comprehensible. It addresses the need for software frameworks that can be used to validate state-of-the-art mathematical models against field observations. Though r.avaflow isn’t fully validated and ready to use in a predictive regime, the general approach and software solutions presented by the authors are a valuable contribution to the community. I do have three major comments and some minor objections, which are detailed in the following.

Major comments

1. You omit including a description of the mathematical model, and rather refer to a publication (Pudasaini 2012). I agree to the first reviewer that this in principle is acceptable. However, at several points in your manuscript you mention that you work with an 'enhanced' version of Pudasaini 2012. It is not at all clear to the reader what these 'enhancements' are! Is it the 'complementary functions' detailed in 2.4.? Please give more details how you deviate from Pudasaini 2012.

The enhancements mainly concern the numerical implementation of the model equations, which is dealt with in the complementary functions (see Table 3 in Section 2.4), a point that is made clear in the revised manuscript. We have also included a functionality to consider the ambient drag (see Table 2 and associated reference) – however, this is not used in the case studies presented here. We have modified the manuscript as follows:

Lines 194–198: Thereby the tool offers implementations (i) of a single-phase shallow water model with Voellmy friction relation (Christen et al., 2010a, b; Fischer et al., 2012) and (ii) essentially the Pudasaini (2012) two-phase flow model with ambient drag (Kattel et al., 2016) and a set of additional numerical treatments (complementary functions) outlined in Sect. 2.4. In the present work we only consider the implementation (ii).

2. To me, the 'complementary functions' block seems to be composed of three groups:

ID 1, 2 and 3 compensate for deficiencies of the numerical scheme. These shouldn't be necessary, if the scheme was shock-resolving, volume and positivity preserving, and well-balanced. You touch upon this in your outlook, when you state that you want to work on the solver in the future. Valuable to the current reader would be a summary of the properties (pro and con) of the current numerical scheme, and if possible, an order of magnitude of the modifications introduced by the 'complementary functions' ID 1-3.

You are exactly right; the deviancies of the numerical scheme appear due to effects that are related to the “real world application” proposed in this paper. The big advantages of the scheme used are that there exists a detailed, fully discrete description in the literature (Wang 2014), and that the scheme served well for various theoretical test cases (e.g., Pudasaini et al., 2014; Kafle et al., 2016; Kattel et al., 2016). However, while the numerical scheme itself should be shock capturing, volume preserving and well balanced (see Wang et al., 2004) we cannot ensure that these properties hold in the practical application (i.e., complex topography). Analyzing the effects of the complementary function is in our opinion not feasible within this paper for two independent reasons:

Firstly, the functions are necessary for real world applications at the current stage of the development of the numerical scheme. A direct, quantitative error analysis (or order of magnitude estimate) is not useful at this point, since the complementary functions mostly prevent “naturally unrealistic” flow behaviour such as excessive diffusion, loss of volume, or oscillations in undisturbed reservoirs. In other words – to apply the scheme to real world problems requires ad hoc assumptions, which are motivated and justified from a process based point of view.

Secondly, to fully investigate this influence an in-depth-sensitivity analysis would be necessary – i.e. how much the choice of model parameters influences/compensates the sensitivity of the model outcomes to the complementary function. To do this in a proper way, it would be necessary to add at least one or two more figures and comprehensive discussion for each of the three functions, and the number of figures is already quite high. We hope that the readers will understand that this is a general introductory paper of r.avaflow.

The reason for introducing the complementary functions 1–3 is outlined in lines *244–246* of the revised manuscript, and the main aspects are included in the discussion (lines *594–610*).

ID 4: It is straight forward to consider entrainment in the model equation (as you also mention in your manuscript). It seems inconsistent not to have this as a part of the numerical scheme. Could you comment on why you chose this approach? And again: is there any additional error expected? Are the update time steps the same for the numerical solver and the complementary function?

The time steps used in the numerical scheme and for entrainment are exactly the same.

The Pudasaini (2012) model does not include entrainment. We agree that the most natural way is to include erosion as part of the model such that the numerical method automatically takes into account the erosion effects (Pudasaini and Fischer, 2016). After the effects are properly described from a theoretical perspective they can be dealt with directly in the numerical scheme – which still requires e.g. full handling of the evolution of the basal topography which is not straight forward: evolution of topography is not a standard transport equation, therefore TVD-NOC scheme automatically introduces diffusion.

To show the potential of using an entrainment model (including the change of basal topography) within r.avaflow we implement it as a complementary function in a first step. We think that erosion aspects can also be included consistently in this way. The results are the same because both will dynamically update the mass and momentum. However, from a practical point of view, it appears that numerically it is easier to consider the erosion and change of the basal topography as it is done now via a complementary function.In the future developments, with improved numerical methods and implementations, we will seek to lift the complementary function for erosion and implement it directly from the model.

The reason for including entrainment in the complementary functions is given in the lines *257–261*. Further, the issue is included in the discussion (lines *611–614*).

ID 5: I like the idea to derive a proxy for the local run-out length. The idea seems closely related to the very common macro-scale approach that relates fall height H to horizontal runout length R (basically stating that the potential energy has to equilibriate work done by friction):

[equation]

A and B being locations on the topography, and xA and xB their projections on a flat plane with R = xB – xA. This boils down to R = H / μ for a block of material that initially has been at rest.

Though your extension to account for initial kinetic energy is straight forward, I am confused by the fact that equation (6) states that material of initial velocity v0 = 0 will have an sstop = 0 (regardless of local inclination and friction coefficient). The situation that you sketch in Figure 3(b) for instance implies a certain sstop. If the slope at which the material comes to rest is steeper that the basal friction angle, however, it should start to flow again in counter direction. How do you account for that based on your stopping criteria?

You are right – the motivation for the stopping criterion arises in close relation to the macro energy approach translated to a sub-spatial-resolution scale. The stopping criterion is only relevant for $v_0>0$. The case $v_0=0$ is trivial and in this case movement will be initiated as long as the local slope is larger than the friction angle. Possible movements in counter-slope (or against the actual direction of movement) are dealt with in the same way – first an imaginary block would have to stop (v=0) and it could start again in the next time step in the opposite direction – the crucial point here is that friction cannot be accelerating, i.e. resulting velocity and friction cannot have the same sign, thus an in between stopping or a more sophisticated time step-stopping control would be necessary. Fig. 3b might be a bit misleading since most stopping probably occurs for small beta | βv | <δ. This is clearly mentioned in the caption of Fig. 3 in the revised manuscript. In the revised manuscript we havefurther added the condition $v_0>0$ (lines *303–305*).

3. Is it possible at all to use r.avaflow with a self-written numerical solver? It would be of very high value if the model/solver could be easily substituted. Can you describe the interface between the numerical method and the GIS framework in more detail?

At the moment you can use the data management and visualization parts of r.avaflow if you write your own solver – the input and output data and formats of the solver, however, would have to be in line with those required by r.avaflow, which are basic asciirasters with a certain nomenclature for the respective names. We acknowledge, however, that this might be rather impractical – so far no interface for distinctly only the solver has been developed, but this could be an interesting future development.

Minor objections

- Abstract and introduction

Please explain what you mean with 'more or less complex process chains and interactions'. My impression is that you mean subsequent events, or events of delayed release time and potentially varying initial conditions. Is that right?

What is meant with these terms is triggering of one event by the impact of a previous event – e.g., overtopping of a lake due to the impact of a landslide into the lake. This issue is clarified in the introduction of the revised manuscript (lines *65–66*).

The more recent work of Iverson (Iverson and George, George and Iverson 2015) provides another two-phase approach for mass flows and would be good to mention. The same is true for GeoClaw which is the corresponding software tool.

Thanks a lotfor the note – in the revised manuscript we have included the work of Berger et al. (2011) and Iverson and George (2016) in the account of the state of the art and also refer to the related software GeoClaw and its extension D-Claw (lines *60–61* and references).

In the introduction you say that none of the models includes the possibility for computing cascaded events. This is actually not quite right. For example in RAMMS you also have the functionality to add deposit onto the topography to study subsequent overflow.

Yes, we have modified the introduction accordingly (lines *92–93*).

- The computational framework r.avaflow

What is the difference between EXPERT and PROFESSIONAL?

r.avaflow [PROFESSIONAL] (in contrast to r.avaflow [EXPERT], which is already explained in the discussion paper) consists in a stand-alone GUI with still reduced functionalities (no parallel processing of multiple parameter combinations, no integrated validation function at the moment) targeted at practitioners. This is made clear in an additional sentence in the revised manuscript (lines *112–113*).

Figure 1: Change of basal topography (s,f,b) is this a typo, or why '(s,f,b)'?

s=solid, f=fluid, b=bulk (explained near the right edge of the figure in vertical letters). b = bulk has been changed to t=total in the revised manuscript.

Please describe your understanding of a 'pixel' and its relation to the surface mesh on which the system is solved. Is the 'pixel' equivalent to a surface grid cell? Or is it rather the projected grid cell?

We do not have surface grid cells in a strict sense (which would be defined by topography following coordinates). With the term ‘pixel’ used throughout the discussion paper we referred to a numerical cell which is always ground projected (GIS coordinates) since we use NOC (see lines *232–240*).

We acknowledge that the term ‘pixel’ might be misleading. In the revised manuscript we replace this term by the term ‘cell’ and explain the exact meaning with the first use of the term.

equidistant quadratic cells: this is a structured, regular grid then?

Yes it is. Please see e.g. line *240* where it is mentioned that the resulting depths are converted into heights for output. An adaptation is applied to utilize the TVD-NOC Scheme which is meant for equidistant quadratic cells.

- Computational experiments and discussion

What criteria / objective functions has been used to decide for the 'optimal parameters'?

The criteria are summarized in Table 4. In fact, an ‘optimum’ parameter value always refers to one specific criterion, an aspect that is clearly mentioned in the revised manuscript.

Figure 11: Is any of these the aforementioned objective function? It is hard to interpret the plots. Can you re-scale the plots to better see the optima?

The functions building the basis for Fig. 11 are given in Table 4. In order to make this clear we refer to Table 4 in the caption of Fig. 11. However, we have decided not to rescale the plots as (i) we would like to show the entire tested range of parameters; and (ii) in those cases where the optima are not well recognizable, they are quite broad and therefore poorly defined (re-scaling would not help here).