

Interactive comment on “GSFLOW-GRASS v1.0.0: GIS-enabled hydrologic modeling of coupled groundwater–surface-water systems” by G.-H. Crystal Ng et al.

G.-H. Crystal Ng et al.

gcng@umn.edu

Received and published: 26 April 2018

RESPONSE TO REVIEWER 2

Boldface: Reviewer 2’s original comments

Italics: Our response

We thank the referee for their time in reviewing our manuscript and providing feedback.

General comments:

C1

Ng et al. present a GIS-based tool, GSFLOW-GRASS, that prepares input and runs the USGS hydrologic model GSFLOW. The authors provide comprehensive description of the GIS-based software, as well as the USGS hydrologic model. The paper is well-organized and well-written. As a modeler, I highly appreciate the authors’ efforts in developing such tools, because “developing inputs to these models is usually time-consuming and requires extensive knowledge of software engineering, often prohibiting their use by many researchers and water managers”. However, I do feel that GSFLOW-GRASS has limited capability in handling spatially-distributed, realistic input data (see specific comments #2). All examples are shown without any measured discharge data. I acknowledge that model calibration is beyond the scope of this study, but it would be helpful if measured data could be shown, to demonstrate that the generated input can yield reasonable, if not accurate predictions. The authors also need to do a better job describing what are the “substantial new/novel concepts, ideas, or methods” in developing GSFLOW-GRASS, as required by GMD.

We are glad that the reviewer found our manuscript to be generally well-organized and appreciated our effort to make integrated hydrologic modeling more accessible. Indeed, we believe model-users are already seeing the utility of our work; just over April 9 to 22 (the maximum length of time tracked by GitHub), our GSFLOW-GRASS repository received 173 views and 22 unique visitors, and one user from a research university sent us an email that opened with “Thank you so much for sharing the GSFLOW-GRASS toolkit. This toolkit really relieves my struggle of preparing inputs.” - and all of this is with absolutely no effort to advertise our toolkit.

After this review, we do now recognize that our original manuscript version failed to adequately explain the novel aspects and new technical advances provided by our toolbox. We appreciate that the reviewer raised this issue, and we have substantially revised the manuscript to address this serious shortcoming in the presentation.

In particular, we first clarified that while some of the individual scripting components

C2

within the toolbox may appear straightforward, our work's innovation is the entire bundled package. Our substantially edited Introduction now emphasizes that existing software for integrated hydrologic models fail to provide freely accessible tool-sets that fully cover pre- to post-processing steps (p. 2, lines 22-30), and that GSFLOW-GRASS addresses that critical gap stymieing the use of integrated hydrologic models (p.2 lines 31-34, p. 3 lines 1-6).

We then explained that the major new method advancement was to create a new set of GRASS-GIS tools that can robustly and automatically generate surface and subsurface model domains suitable for hydrological modeling, which was critical for GSFLOW-GRASS to be widely applicable to a diverse range of hydro(geo)logical settings. We now realize that the original manuscript version documented these new GRASS GIS extensions but provided almost no background on the challenges of creating robust and automated tools, which has led to a general unavailability of such solutions predating our toolbox. A new paragraph has been added to the Introduction to present this technical advancement (p. 3 Lines 7-21). Further, we have entirely re-written Section 3.2 on the GRASS GIS domain builder (p. 10-12), so that it now explicitly describes what was implemented to solve specific known challenges for stream network delineation. Finally, we also made major changes to Section 4 on the Examples, in order to explain how each example demonstrates a different strength and capability of the domain builder (specifically, p. 21 Lines 3-10 for Shullcas, p. 21 Line 32- p. 22 Line 4 for Santa Rosa, and p. 23 Line 10- p. 24 Line 4 for Cannon River). These examples demonstrate how GSFLOW-GRASS handles known challenges with various degrees of drainage integration, landscape relief, and grid resolution, as well as the presence of irregular coastal boundaries.

The new technical advancements of our GRASS GIS tools were recently highlighted as a new release feature on the GRASS GIS website: <https://trac.osgeo.org/grass/wiki/Grass7/NewFeatures74> (including a figure with our Cannon River watershed example) - see screenshot in Figure 1.

C3

In addition to major revisions to the Introduction, GRASS GIS Domain Builder section, and Examples section, we essentially re-wrote the Abstract and Conclusion to highlight these new and technical contributions.

The reviewer also brought up an excellent point about the importance of spatially distributed, realistic input data. This prompted us to add new capabilities to the toolbox to accommodate spatially variable inputs, which we describe in greater detail under "Specific comment #2."

We appreciate the reviewer acknowledging that model calibration is beyond the scope of this work. However, the reviewer's comment about comparing against observations made us realize that we needed to be clearer that the aim of the examples is not simulate realistic results for each site but is instead to demonstrate the robustness of the domain builder for a range of settings (as pointed out above) and the types of processes that can be explored with GSFLOW-GRASS. We re-wrote the opening of the Examples section to reflect this (p. 18 Line 32- p. 19 Line 5), and we added a brief discussion of how the automated GSFLOW-GRASS toolbox jump-starts the model implementation and thus facilitates additional calibration and sensitivity analysis (p. 19 Line 18- p. 20 Line 7).

Although the examples are not meant to serve as realistic results, we do recognize the value of comparing our example simulations against actual observations. The Cannon River site is the only one of the 3 example sites that has publicly available discharge data, and so we now show it in our revised Figure 7. In addition to discussing where the uncalibrated model does reasonably match observations, we also point out how the major discrepancies found in the default implementation can be useful for guiding parameter calibration (p. 24 Lines 9-15).

Specific comments:

1. Besides GSFLOW-GRASS and Gardner et al. (2017), there is another for-free

C4

software by Earthfx that can generate the inputs for GSFLOW. It is surprising that the authors did not review or describe this software. What are the differences between GSFLOW-GRASS and Earthfx software?

We were aware of the Earthfx software and did mention it in the original manuscript ("However, with the exception of a single for-fee software product by Earthfx (<http://www.earthfx.com/>), these tools neither help users conceptually link the different process domains represented in PRMS and MODFLOW nor generate the files that link them." original p. 2 Lines 21-23). The reviewer wrote that Earthfx is "for-free" but we wonder if that is a typo and is supposed to be "for-fee"? Looking again at the Earthfx website, we do not see anywhere that one can freely download the software (<http://www.earthfx.com/VIEWLOG/Components/VLGSFLOW.aspx>) and only see a page pointing interested users to call for pricing information (<http://www.earthfx.com/VIEWLOG/Details/Pricing.aspx> says: "Call for pricing," "University discount 30%" for the VIEWLOG software, which includes VL-GSFLOW).

Given that the software is fee-based and cannot be accessed without contacting the vendor, we found it difficult to review in detail. We did add to the text that Earthfx offers "full support" for GSFLOW, specifically through its "VIEWLOG" software package (p. 2 Lines 25-26). We then point out that "the community still lacks a free and complete package spanning pre- to post-processing for heterogeneous surface and subsurface domains" (p. 2 Lines 27-28), motivating the development of GSFLOW-GRASS.

2. The GSFLOW-GRASS tool has limited capability in handling spatially-distributed, realistic input data. For example, P10, L24 "In its current form, v.gsflow.segments ... allows the user to set a single channel width and Manning's n (in-channel roughness coefficient for flow resistance) across the whole domain;" P13, L29 "we do provide a script for uniformly applying a single climate data series over all HRUs to create climate_hru files;" and P14, L24 "most parameter values in printPRMSparamfile.py are preset. . . This includes various soil

C5

and land-cover inputs, such as soil_type, cov_type, transp_end, and pt_alpha." There are GIS-based hydrologic model input tools that can take all different types of GIS input to generate spatially-distributed input data from national data-base or in situ measurements. For example, PIHMgis (Bhatt, G., Kumar, M. and Duffy, C. J., 2014: A tightly coupled GIS and distributed hydrologic modeling framework, Environmental Modelling & Software. 62, 70-84.). The spatially-uniform approach and the preset default parameter values may prohibit GSFLOW from generating accurate predictions.

Our original GSFLOW-GRASS version did include an option for spatially heterogeneous hydraulic conductivity, which was implemented in the Santa Rosa Island example. However, we agree with the reviewer that the general lack of ability for our original toolbox to generate spatially variable data inputs (beyond hydraulic conductivity) was a weakness in the original submission. To address this, we modified the toolbox to accommodate heterogeneous channel width and Manning's n parameter through the Settings file (as explained on p. 9 Line 29 and on p. 17 Line 5). For other inputs, we created a new GRASS GIS tool, v.gsflow.mapdata, which can take any spatially variable data in a raster or vector GIS format and map it to one of the GSFLOW discretization structures: sub-basin HRUs for PRMS surface-water processes, regular grid cells for MODFLOW groundwater processes, gravity reservoirs that link the HRUs and MODFLOW grid cells, or stream segments or reaches for MODFLOW streamflow processes. This allows users to incorporate data from any source to the GSFLOW-GRASS data structures for input into the model. The new v.gsflow.mapdata tool is presented on p. 9 Line 30- p. 10 Line 8. Throughout the rest of the paper, we also mention how this tool can be implemented to create specific spatially distributed inputs, including the climate inputs and soil / land-cover parameters mentioned by the reviewer (p. 12 Line 26, p. 14 Lines 19 and 29, p. 15 Lines 16 and 29, and p. 17 Line 1).

For linking integrated hydrologic modeling with existing databases for model inputs, we do see the clear utility but consider this to be beyond the scope of our current work,

C6

which aims to provide a generalized solution for implementing GSFLOW-GRASS. We reference software tools that focus on this capability, including Bhatt et al. (2014) as suggested by the reviewer (p. 10 Lines 3-5); we then point out that these databases are typically only available in observation-rich places and thus we do not include it in the first GSFLOW-GRASS version, which provides a general basis for further development (p. 10 Lines 7-8). Our revised conclusion discusses future extensions of GSFLOW-GRASS to include links to spatial databases to generate model inputs (p. 25 Lines 11-12).

3. Page 3, L10 The authors state that triangulated irregular networks have better water balance performance. Why is that?

We realized that we left out some details and should have specified that triangulated irregular networks (TINs) show better water balance performance IF they are implemented with the finite volume method (because the finite volume method is mass-conserving), and that TINs cover complex surface domain more efficiently (fewer units) than grid cells. We edited the text to say all of this on p. 4 Line 15-16.

4. I am interested in the spin-up process described between L6 and L11 on Page 9. The authors describe the initial conditions as preliminary steady-state initial conditions. Usually the spin-up process is aimed to bring models to steady-state. If so, what is the difference between the before- and after-spin-up initial conditions?

We realize we could have explained the steady-state vs. spin-up process much more clearly. The preliminary steady-state run is just for the MODFLOW (groundwater) component; the steady-state groundwater head results are then used to initialize the spin-up for the fully coupled (surface AND subsurface model domains). We edited the text to better explain this on p. 10 Lines 13-17.

C7

5. Some of the parameters are shown without their definitions, for example pref_flow_den and sat_threshold. It would be add definitions.

We added Table 1 to define all GSFLOW parameters mentioned in the manuscript.

Technical comments:

1. Figure 2 caption “Duncan runoff and fast interflow occurs in the preferential-flow reservoir.” Should be “occur.”

We implemented this edit.

2. P10 L4 “This approach is complementary to the grid-cell HRU approach of (Gardner et al., 2017).” \citet command should be used instead of \citep.

The text was edited in the revision and no longer has this issue.

3. P22 L8 “This allow users” Should be “allows.”

We implemented this edit.

Interactive comment on Geosci. Model Dev. Discuss., <https://doi.org/10.5194/gmd-2017-321>, 2018.

C8

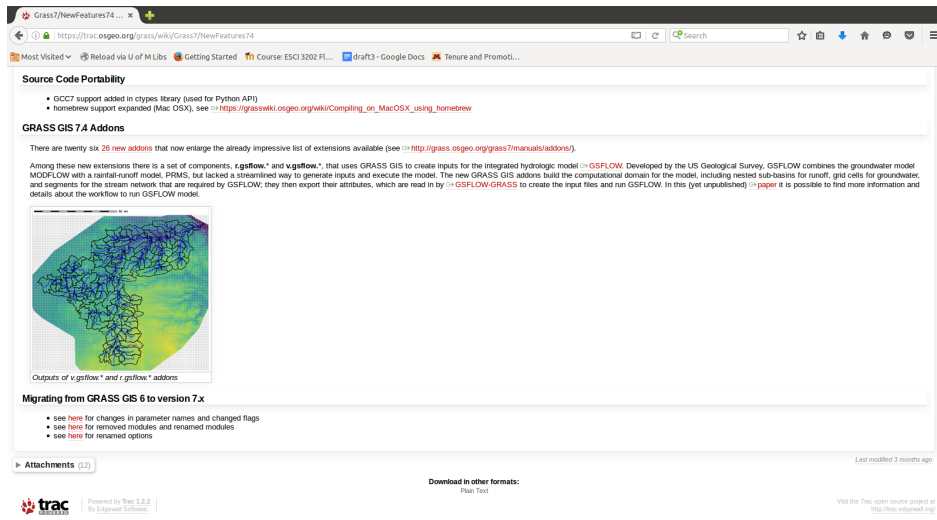


Fig. 1. Screenshot of GRASS GIS' new features - includes GSFLOW-GRASS add-ons