Earth **Surface**
Dynamics
Discussions

Open Access

ESurfD

Interactive
comment

Interactive comment on "Computing water flow through complex landscapes, Part 3: Fill-Spill-Merge: Flow routing in depression hierarchies" *by* Richard Barnes et al.

Daniel Hobley (Referee)

hobleyd@cardiff.ac.uk

Received and published: 29 May 2020

This paper presents an efficient, novel method for the routing of flow across depressions while still honouring the mass balance of the water flux. This is an important computational geomorphic problem of long-standing, and this submission is certainly worthy of publication in ESurf. I would expect the algorithm to be taken up widely by the surface process and hydrological modelling communities.

I should note that I have not reviewed parts 1 and 2 in any great detail; this review focuses specifically on this as part 3. I will leave it to someone higher up the editorial chain to ensure there isn't too much overlap here, but based just on this manuscript, it

Printer-friendly version

Discussion paper

doesn't seem like there will be.

Thanks to the install issues I've had (see technical comments), I haven't been able to actually run the software. However, once installed cleanly, I imagine this will work fine. Happy to quickly confirm after corrections have been made on the manuscript – but thought you'd rather not have the delay on getting these comments while we sort annoying OS and versioning issues on Github.

I like this manuscript a lot, and can recommend it be published with what I think are minor and hopefully pain-free revisions. I think that the only comments I have that would involve substantial changes to the work are around a. being more clear in describing the algorithm, and b. adding a few synthetic performance tests.

I have provided some general comments, some line item comments, and some technical comments (i.e., on the code itself) below.

I look forward to seeing this in print in short order!

Dan Hobley

—

General comments on the article

In general, this is an extremely well-written manuscript, especially considering the audience likely to see this paper in ESurf and the level of technicality that is clearly actually involved here under the hood. I've pulled the following broader comments out because they don't have specific linenumbers, rather than because they are particularly major.

* I found the way the depression hierarchy is talked about in the Algorithm section quite confusing. I think the reason is that the authors convolve the object that is the hierarchy itself (i.e., the input to this algorithm), the conceptual idea of the DH, and the method used to build that hierarchy (as explained in Barnes et al. (2020)). For example around P5ln30 "The specific outputs from the depression hierarchy [the method] that

are used in the Fill-Spill-Merge algorithm are: – DH: the depression hierarchy [the object] itself…."; P8ln2 "By routing water according to the DH…" -> so is this the object, or the method? Finding a distinct way to refer to the method/algorithm that produces a DH, the ordering that is described by a DH, and a DH structure itself would significantly enhance clarity throughout. (See line item P9ln26 also.)

* Jargon suddenly appears at the end of p.9. The whole section P9ln25-P10ln17 I found very confusing. I ramble about this at length in the line comments below, but at heart this needs a. the jargon explaining, b. the text simplifying, and c. the text expanding as necessary.

* Regardless of that, I would strongly advocate that you put some actual pseudocode into the paper – maybe as supplemental material, but in the body would also be good. This would really help in sections like that one, and the one to follow (both noted below).

* Similarly, if iterations like those on P10 are described, then I think they would be much clearer with cartoons illustrating the possible steps next to them. You use fig 2 as a case study for the sequence on P11, but it would be much clearer to produce a second version along the lines of fig 2 but with multiple parts. These would highlight each inner loop as described in the text, what happens under the possible configurations of that loop, and the sequencing of the water filling in under the rules for those configurations.

* Oceanlink-geolink terminology. I like the conceptual framework of the paper a lot, but the description of "paired oceanlink-geolinks" I found confusing. Because there can never be an oceanlink without a geolink, I would tweak the terminology such that an oceanlink is a subset of a geolink, and the pair concept goes away.

* The maths as described on P13 is not strictly correct. The equations give summed elevations, but they are variously described as "capacity" or "volume". There should be cell areas in here somewhere. Also, there is a missing definition of N. Check the maths again for precision of terms.

* Section 5 – The text asserts that O(NlogN) is the best fit, but the paper does not show it. At the very least, plot the data in Table 1. . . .however, doing this will reveal that the data points do not in fact give a good smooth correlation, convincing of NlogN (which is presumably why you didn't do it in the first place!) I would retain Table 1 as nice examples of "real world" model performance, but add a synthetic test with random noise elevation on varying size grids. This should give you the convincing NlogN scaling to plot up and push a NlogN scaling best fit through, and is also a cleaner – and more standard - way making the point about actual performance of your implementation. Some light restructuring (see below) would help here too.

* You could add a statement somewhere that this algorithm also permits water to be re-tained in the depressions between runs, if that is desired – so you can get a timeseries as the lakes fill over time. It would also easily let you perform true water balances on the lakes if the user wanted to, by subtracting water between steps. This is implicit in the text, but it's a cool feature you could emphasise a bit more if you want.

—

Line-item comments:

P1ln16 – misformatted ref

P3ln15 – 0 seems a special case here. Refer also to 12 or 11?

P5ln10 – The ocean links. You should make it clear here why, from a modelling per-spective, you need to separate ocean links from geolinks, so we know where we're going. Also, you have inconsistency between "ocean link" and "oceanlink" – check for consistency (incl. In figure captions).

Fig 4. This caption could be a lot more generous to the reader. Explain the colorcoding; what *exactly* the blue shading is; and exactly what you mean by "non-additive". I don't quite follow, and feel there must be a clearer way of expressing this. Also, would it not be clearer to actually let area represent volume of water? (I assume that this is not the

case is what you are getting at by "non-additive"?) In particular g-j are quite hard to follow and need more text. Isn't j inconsistent with f? You appear to suggest in f the whole system floods out, but j does not show this. Format-wise, you have a,b,c on the fig and A,B,C in the text.

P9ln10 – "depth/breadth-first traversal". Please explain what this is in plainer English.

P9ln25 – Suddenly, the paper becomes much more jargon-y and hard to follow. Clarifying depth-first at ln 10 will help, but what is "post-order"? "an invariant"? Please explain these in plainer English. This paragraph and the next are also quite confusing aside from the jargon, so please expand and be more generous with the text as well. Things I am struggling with: How does a depth-first traversal have an "outer" rather than a "deeper"? Clearly the water starts in the leaves, so how can the outer (first??) traversal begin in the ocean (ln 30), which is a trunk? How are the traversals nested, and when do we shift from outer to inner and vice versa? An explanation of the jargon in a pre-paragraph will probably sort most of this – i.e., what are outer and inner traversals, what is pre- and post-order, and how are these concepts interrelated? Another flow-chart, a diagram, or chunk of pseudocode would also really help here. The steps on P10 help piece together what these paragraphs must mean, but they come after the confusing text not with it, and the text should really stand on its own.

P10ln10 – "We add water *from A* to B"? Where does the added water come from? Is it the original volume or the passed volume? (Must be the latter, but this is not as written) Tweak the phrasing here.

P10ln11 – "this part of the algorithm". Do you mean point 1, the inner traversal, or the section 3.2? Point (iii) in this list is also very confusing. Break this case out and say what you mean (and why this is a terminating case) more clearly. The manuscript has not described the concept of an oceanlink-geolink pair (surely the link between two nodes is either an oceanlink or a geolink, so there can be no pairing?), so clarify that too (see main comment). Again, this feels like a diagram or pseudocode or both would

Interactive comment

C5

add a lot here. The diagram I have in mind would show the possible cases for this 1-3 loop as a sequence of filling lakes, as in Fig 2 or fig 4a,b, g-j (but where water volume is honoured). Although I appreciate this section looks like this because it explicitly mirrors the structure of the code, there is surely a clearer way to express 1-3. E.g.

– 1. If a node has excess water, attempt to pass that water to an unfilled sibling.

– 2. If no unfilled sibling is available, pass the water to the parent and terminate inner loop.

– 3. If no parent exists, the node must have an oceanlink, so pass the water along the oceanlink and terminate inner loop.

– 4. If the oceanlink ends in the ocean, discard the excess water and terminate inner loop.

Does this not encompass all the logic without the detail in the subcases, the pseudo-parents, etc.?

P10ln31 – apart from still disliking the "paired" terminology here, this also brings back the pt (iii) I didn't understand above. This makes it clear what you meant there, but the idea that 12 is sort of a parent of 4, but not really, and only sometimes, is a confusing one. I would stick with the idea that parents are strictly in the same tree, and replace this idea with simply "passing water down the oceanlink to move it between trees". I don't see what else it is adding for the cost of being confusing. In general though, this stuff on P11 is really good.

P12lns3-4 – I don't see why the algorithm needs to start at the ocean and iterate back to every leaf, only to iterate forward again. Given we already maintain a list of all the leaves, associated pits, and labels, why can't me just start at the leaves and only go forward? If I'm missing something, it means this needs clarifying anyway. Are you trying to say, iterate from the ocean down along all the branches, but pinch out the branch whenever you find water and don't go further?

P12lns15-28: Another place where a block of pseudocode alongside the text would add a lot. Arguably it could replace the text entirely, but having both would be extra clear.

P12ln29. "To do so, we imagine a hypothetical outlet that drains the depression. . . add:"at this elevation". This section for me says the same thing three times in different words. I would have simply: "Step 1 in this approach requires an efficient way to determine the volume of a depression below any given elevation. To do so, we imagine a hypothetical outlet that drains the depression at that elevation. If we call the elevation of this hypothetical outlet o and. . ." (also see comment below about maybe removing the idea of a hypothetical outlet entirely.)

Equation 1: You don't actually define N

Equation 3: I don't object to it, but why is there a bar through V?

P13ln4 – Run-on; end with a period not comma or tweak phrasing.

P13ln5 – "the sill". Which? The hypothetical one, or the real one?

In general here, I think the concept of an "imaginary" outlet at every possible node height is getting in the way of what you're trying to say. Isn't it just that you can track the volume of a depression below various depths using the equations you give by cumulative summing along the plain queue, and then can compare that increasing volume as you move along the queue to the water volume available, which tells you the fill level? I would rephrase the way you present P12ln29-P13ln9 without this imaginary sill/outlet concept.

P13ln21 – I don't think this is the right heading for the section. Isn't this "Algorithm performance" or something like that more specific? 4 and 5 would then be subsections, e.g., "theory", "computational performance". The section itself reads really well though – and I assume by this point you will have defined the iteration jargon a bit better at first use.

Interactive comment

Printer-friendly version

Discussion paper

P14ln32 – If you rename the section, I don't think this final sentence is needed.

P15ln11 – consider "Applications and testing on real-world data" or words to that effect (assuming you renamed 4 above, and remove the subheading below).

P16ln1 – there is no 6.2!? Remove the subheading. Otherwise, I like this section a lot.

...overall, I think a light restructuring (i.e., broadly just some section renaming and text shunting) through 4-6 would make this a bit stronger. What you have is testing of performance on both synthetic (please!) and real-world data, then a real-world case-study that attempts some model validation. I think the sectioning should reflect that a bit more explicitly, e.g. something like,

4. Model performance

– 4.1 Theoretical analysis

– 4.2 Performance on synthetic data

5. Field testing and intermodel comparison

– 5.1 Performance on real world data

(brief chat about Table 1 to give people an idea of general performance on real-world data, or just remove and junk these subsections)

– 5.2 Model intercomparison

(This is basically just 6.1 – and by all means, keep the performance data for this case study within this section)

Conclusions do the job well.

—

Technical comments

Install instructions in the readme require cmake – and so should probably explicitly

direct the user to acquire this first. I note configuration issues are also likely, which should probably also be clarified (see below); if they are not, the authors are dramatically reducing the potential reach and impact of their work. In general, I think the readme needs expanding a bit (see technical comments).

I severely struggled with the install for the software, and am probably significantly more tech-savvy than many target users. Did the authors test installs thoroughly against a full range of OSes? I am running macOS Catalina, XCode 11.4 (i.e., fully modern installs) and anaconda, and am seeing a spew of cmake errors which some investigation reveals are related to incompatibilities between the installer, the shipped SDKs with XCode, and anaconda. This is going to be a common user case, and I think the authors need to address it, if only briefly, in the docs – and test some other common environments too for safety's sake.

[Part of the answer to my problem is here: https://www.anaconda.com/blog/utilizing-the-new-compilers-in-anaconda-distribution-5. The install process isn't compatible with the most up-to-date XCode, and the user needs to get the 10.9 SDK off Github as described there, then point the compiler at it with "export CONDA_BUILD_SYSROOT=/Opt/MacOSX10.9.sdk" (assuming it's downloaded to Opt).]

(I should note the above broadly duplicates an issue I have raised directly on Github.)

The current install instructions appear to assume GDAL is installed on the target machine, and do not say so. They must make this explicit, as the triggered install errors are extremely opaque. I gather from asking on Github that these errors do not actually prevent install, but the readme needs to provide that guidance because the errors look pretty scary – and also should I think recommend a GDAL install anyway. [I got GDAL in a pain-free fashion using homebrew on my Mac (https://brew.sh), if that helps – or just let users figure it out for themselves.]

There should also be a line in the readme telling the user how to launch the software

after install, as there is over at r-barnes/Barnes2019-DepressionHierarchy.

My familiarity was C++ testing is not very complete, but I don't believe this code has a *formal* testing framework attached to it. This is more an observation than anything else, but I would informally recommend that the authors consider building one at some point in the future. This is the way the wind seems to be blowing in research software development at the moment, and it might be good to get ahead of the pack on this.

On the basis that the code is not formally tested, I have not attempted to extensively review the code itself. However, informal tests are applied in the manuscript and in the code (the correctness tests), and this for me meets the basic requirements for a submission like this. Could you add a line in the readme telling the user how to launch the correctness tests for themselves...? This would also serve as a nice example of the code doing its thing.

As claimed in the manuscript, the code is indeed well commented and neatly and nicely structured – good job!

Code availability statement is great – but you should consider actually putting an open source licence onto the code and making that explicit here (after all, this is already de-facto open source code, so you should probably formalise it). E.g. add a LICENSE.txt and add "The code is released under an MIT Open Source license [or your preferred license]" at the end of this section. But your call.

Interactive comment on Earth Surf. Dynam. Discuss., https://doi.org/10.5194/esurf-2020-31, 2020.