Earth Surf. Dynam. Discuss.,
doi:10.5194/esurf-2016-45-AC4, 2016

Earth Surface Dynamics Discussions

ESurfD

Interactive comment

Interactive comment on "Creative computing with Landlab: an open-source toolkit for building, coupling, and exploring two-dimensional numerical models of Earth-surface dynamics" *by* Daniel E. J. Hobley et al.

Daniel E. J. Hobley et al.

hobleyd@cardiff.ac.uk

Received and published: 24 November 2016

Dear Benjamin,

Many thanks for completing this review for us. As requested by the AE, please find below our replies to your comments. We've pasted each comment you made, then replied to it, and as necessary then copied in any modifications made in the text. We hope that these replies address your concerns to your satisfaction.

I also note that the process of pasting this text from Word to the ESurf online interface has resulted in our carefully formatted bullets now appearing as little inline "nu" symbols. Sorry! Hope this isn't too annoying.

Thanks again,

Dan (on behalf of all the authors)

Review of Hobley et al. creative computing with Landlab

With Landlab, Hobley et al. present a refreshing framework to integrate numerical solutions for planetary surface processes. The model is fully open source and targets both end-users and developers providing an extended library of model structures and implementation strategies. The wealth in existing models and model structures makes the paper timely for the geophysical earth surface community and I recommend the paper for publication after consideration of some remarks.

Before reading the paper and because of its intention: i.e. providing an integrative numerical modeling toolkit, I formulated some questions which I would like to see addressed in such a paper. In the following I list those questions and discuss to what extend I found satisfying answers and where the paper can be improved.

* General ease of use of the numerical model. I congratulate the authors for their well-functioning and carefully documented source codes. ν Thanks!

* Debugging tools I highly value the availability of unit tests to evaluate the model code. This is one of the aspects in which Landlab differentiates from other numerical earth surface models. ν Again, thanks. Good to be reassured that effort on this front has been worth it...!

* Input/output data structures. Compatibility with other tools available. Well implemented and properly discussed in the paper ν Thanks.

* Discretization of the continuous solution to grids. Shifting between different kind of grids. Another major contribution of the paper is the way in which different discretization schemes and the numerical grids which come with them are presented. Nonethe-

less, while reading section 3.1, I sometimes had the feeling reading technical notes rather than a paper on earth surface dynamics. I propose that the authors discuss, somewhere in this section, the importance of different grid implementations and its relevance for the field of earth surface dynamics. E.g. the use of structured grids has major implications in terms of landscape symmetry as elegantly documented by Braun and Sambridge (1997). It would also be interesting to briefly discuss the implications of using raster versus voronoi grids and to what extent the model supports conversion between the methods. A first example which comes to mind in the framework of LEMs (which is only partly what the model is designed for) is the simulation of tectonic shortening which is often executed on irregular grids (Willett, 1999), whereas other landscape processes such as nonlinear diffusion (Perron, 2011) might benefit from structured grids for computational performance. Does the grid structure of the model allows for grid refinement (e.g. Künze and Lunati, 2012)? ν The first of these points is a good idea, but we feel we've already done this fairly well in the manuscript. The paragraph now at p. 8 lns 17-24 covers this ground at a level we already thought was commensurate with the scale of this paper, and indeed does already briefly touch on Braun & Sambridge's insights. Nonetheless, we have added an additional clause with a little more detail from Braun & Sambridge ("Irregular grids avoid some of the cardinal direction artifacts than can form on regular grids, such as linear networks and linear drainage divides, as well as consequent biases in measured channel metrics like drainage density, river length, and channel slope (Braun & Sambridge, 1997).") Regarding the later comments here, however, we are reluctant to expand too much on what we have regarding explicit conversion between grid types. Landlab is not, and is not really at this time ever intended to be, a GIS tool. Operations like interpolation between grids are better accomplished in other, more specialised software in our view. At the moment, in part for similar reasons and also simply because of the lack of a pressing application, we do not allow grid refinement or densification. Past experience of some of the Landlab team (GT, NG) with CHILD – which does support this kind of ad-hoc grid modification in some instances – suggests that this kind of process can be extremely challenging to implement. This would only be more so in Landlab, where densification would need to be accommodated not only within the grid, but its consequences also propagated out into many of the components. Nonetheless, we do intend that future versions of Landab may explicitly support grid deformation, if not refinement per se. That all said, we should probably also note that specifically regarding comparison between results obtained from different grid types, this is an active field of research for the Landlab team (see, e.g., Gasparini et al., Eos Trans AGU 2014, EP51E-3564), and future publications will address this issue specifically.

* Parallelization and suitability for supercomputing One essential asset of a robust model structure, especially if the authors target a large user community, is the possibility to port the source code to a larger computer infrastructure. This item is currently not covered in the paper and it would be good to discuss on the potential of Landlab to be parallelized and the limitations which comes with it. In case the model supports parallelization, how do you take care for changing drainage areas while executing the model on different computational blocks? ν Again, this material was not present in the paper primarily for reasons of space in what is already a long manuscript. As you are of course alluding to, parallelizing a heavily componentized software architecture is significantly more challenging than a fully compiled and stable code, as the programmer can not necessarily know ahead of time the sequencing of the calls to various parts of the code. This more or less rules out the idea of "parallelizing Landlab" in the broadest sense. However, parallelization is much more possible within individual components – and is implemented in Flexure, as proof of concept. However, given your explicit interest, we've added a short additional section:

"3.3.3 Parallelization

Together, the componentized nature of Landlab and the level of flexibility afforded to the user conspire to rule out the idea of Landlab as a whole being highly optimized through parallelization. However, there is great potential for parallelization of Landlab at the component level, since the run methods of each component are entirely selfcontained. As proof of concept, the Flexure component has already been parallelized (see online code and documentation). Although in Landlab version 1.0 we have not had a compelling enough use case to invest significant time in such work, many of the components already in the library would be amenable to parallelization in this style, and this could be done in future releases."

* 3 spatial dimensions Many earth surface processes require a 3-dimensional spatial discretization. Amongst others: hydrological processes, ice dynamics, terrestrial heat advection and diffusion, and soil dynamics. As the authors explicitly state that Landlab "is not a landscape evolution model ...rather, it presents a framework under which a wide variety of models can be implemented using it tools, including hydrologic ...", I am a little surprised that the use of a third spatial dimension is never mentioned in the text. I do not suggest that the 3rd dimension should be included in the current release of the software but I would find it interesting to see a discussion on the possibility of adding a third spatial dimension to the model. ν I'm assuming here that you are simply advocating that we discuss whether Landlab be able to run three dimensional grids. I don't feel this is necessarily close enough to the remit of Landlab (which is at the moment, fundamentally, a two-dimensional framework), as we set it out in the introduction, to be worth spending time on in the manuscript.

* Numerical accuracy. During last decades, many numerical models have been developed mainly focusing on (i) earth surface processes being simulated or (ii) the performance of the numerical simulations. Much less attention has been given to the numerical accuracy of the developed models. This issue is also not very well covered in this manuscript. Nonetheless, recent work has shown that numerical accuracy significantly influences model performance, not only when topographical knickpoints are present but also when lateral displacement of topography needs to be accounted for (Campforts et al., 2016; Campforts and Govers, 2015). Moreover, it would be good that Landlab also offers some analytical solutions for the numerical processes being simulated or that the authors discuss how this could be achieved as analytical solu-

tions are the benchmark against which different solution strategies can be evaluated. As such, a benchmark solution would be available which cannot only serve as a tool to evaluate the accuracy of the current model structure but is also of aid to new users in order to test new models and numerical solutions. For example, an analytical solution for the stream power model can be quite easily found using the slope patch solution of Royden and Perron (2013) but also for numerical solutions not directly related to landscape evolution, analytical solutions do exist (Stüwe et al., 1994). Numerical accuracy is sometimes mentioned throughout the text (e.g. when discussing the influence of timesteps on line 30, p. 15) but it would be good to elaborate on this point and cover this relevant issue in more detail ν As you note, this manuscript does not dwell on issues of computational accuracy. However, this is mainly because this text is focused on the functionality of Landlab as a whole, whereas issues of numerical accuracy are relevant to each component individually. On this basis, we would prefer not to greatly expand our coverage of this issue; rather, it will be discussed as relevant in future publications which are focused on each novel component in turn. For instance, Adams et al., in review, doi: 10.5194/gmd-2016-277 does this in some detail for the implementation of the overland flow model in section 5.4; a manuscript in prep from Siddhartha Nudurupati on the vegetation CA seen in section 5.3 does something similar for that component. [I should also note as an aside that the Adams et al. manuscript is now citeable in review at GMD, so we have reinstated references to it in the text as appropriate.] Likewise, the diffusional and stream power models illustrated in sections 5.1 and 5.2 are simply reimplementations of algorithms already comprehensively described by other authors; existing publications describing those schemes specifically (c.f. Braun & Willett, 2013 for stream power; e.g. Slingerland & Kump, 2011, Chapter 5 for discussion of central differencing schemes for diffusion) already provide much of this information, and we assert that it is not necessary to repeat here.

* To what extend is the model oriented towards integration of measured data into the numerical model? Major advances can be made in the field of earth surface processes by combining numerical models with field data (Fox et al., 2015; Glotzbach, 2015). I

think the framework presented in this paper offers an excellent opportunity for such an approach and could be of high value for otherwise more data oriented researchers. It would be interesting to see some explicit suggestions or remarks on how to combine the Landlab framework with field data. ν In the interests of space and keeping the manuscript focused, we chose not to include this material. We should also re-emphasise that this kind of functionality will probably appear at the component level, so would be best addressed in future, component-level publications. We anticipate that there will be a good number of these, and indeed, forthcoming publications from the core team (e.g., Siddhartha Nudurupati et al., in prep; Hobley et al., in prep) are already making strides in this direction.

* Particle tracking One very interesting aspect of numerical models is to trace back particles through time. I am wondering whether Landlab allows fingerprinting of for example sediments. Is it possible to track a sediment particle all the way from the hillslopes to the outlet of the drainage network? Can this be done for all particles and how feasible is that in terms of memory allocation and data storage? Particle tracking would offer great potential for the model to be further combined with e.g. detrital dating methods (thermochron data (Herman et al., 2015)/ CRN data (Mudd, 2016))). ν This is indeed an interesting aspect, and we very much intend for future components to include a particle tracker. However, given that (as Wickert also noted) the current suite of components at version 1.0 is not particularly sediment- or deposition-orientated, we haven't prioritized this functionality, and it seems unnecessary to speculate on it here without having the work in hand.

* An interoperability interface

In the abstract, Landlab is presented as a modular framework with a strong interoperability. The paper comes up to this promise and presents four contrasting model designs. I enjoyed reading these sections and I think they strongly contribute to the illustrative nature of the paper. Nonetheless, I think this section could be even stronger if the authors also point out the advantage having the different model approaches under the roof of one single model infrastructure. In my opinion, one of the main assets of Landlab is not necessarily the capacity to execute separate model designs but exactly that they can be combined. For example, until now, little attention has been given to the role of vegetation in landscape evolution models. Nonetheless, it has been shown that vegetation might have a very strong influence on landscape evolution (Collins, 2004). The cellular automaton presented in sections 5.3 could therefore be neatly integrated with section 5.1 and/or 5.2. Likewise, hydrological processes are shown to govern stochastic processes like shallow landsliding (Montgomery and Dietrich, 1994). Hence section 5.4 can be perfectly linked up with the other sections. I would keep the sections in their current form as they are very illustrative of the model performance but it would be definitely nice to discuss and eventually illustrate the potential to integrate all or some of these modules. ν We had hoped that this point was coming across strongly enough already, but we're happy to address this a bit more forcefully. We've added "We hope that these examples will also serve as an illustration of the potential power of the Landlab framework to enable novel or under-explored process interaction studies (e.g., of vegetation on landscape evolution; of surface hydrology on stochastic surface processes)"... to the introduction of section 5.

Minor comments

Overall the paper is very fluently written and I enjoyed reading it. I endorse the comments of W. Schwanghart and A Wickert. In addition, I have a few minor comments: Page 2-5: Introduction. I get the point the authors try to make but the way the introduction is written feels like the authors want to promote their product as the one and single framework to be used for future model development. Although I agree with the authors that the structure they provide can offer excellent guidance for new model development and the 'gridding engine' can definitely save many hours of numerical coding headache, I am not so sure whether I agree it is a good idea to demotivate building code from scratch. The latter is still the best way to become familiar with the implications and limitations which come with numerical software design and to realize to power and caveats of numerical models. I second W. Schwanghart's comment in suggesting that many open source initiatives exist, although sometimes written in commercial software (e.g. TopoToolbox and the LEM TTLEM is fully open access; SIGNUM (Refice et al., 2012)) or even in Python, advanced GIS packages are available (LSDTopoTools https://github.com/LSDtopotools). ν We aren't sure what exactly we can change here (outwith our modifications pertinent to Wolfgang's comment). It is certainly not our intention to demotivate future readers to code up stuff for themselves! That said, it's clearly the purpose of this manuscript to set out Landlab's shop stall. We've made a couple of minor tweaks to the language used to emphasise more that the issues described here pertain specifically to issues for professional research scientists who just want to produce high quality software, and don't necessarily carry over to considerations around learning or skill acquisition (See track changes in paragraph 2, p. 2). Hopefully a future publication in a different forum will expound on Landlab's advantages as a pedagogical environment.

Page 10, line 24: Analytical method. I would use a different wording as analytical typically refers to the analytical solution of e.g. PDE's. What about grid methods? ν Good call on changing this. We've selected "Computational methods".

Page 15-16, line 23-5: I find this very interesting. Does the model allow for constraints on the timestep in the case that implicit methods are being used? Although they are indeed unconditionally stable, the use of large (main model) timesteps in LEMs might result in in very sudden topographical changes causing the presence of artificial steps in the landscape. E.g using an uplift rate of 1mm/yr in combination with timesteps of 10ky results in a sudden uplift of 10m in the LEM which initiates artificial knickpoints at the baselevel of rivers. ν Aha, good question. As is implied (though I don't think outright stated) in this section and its associated figure, when Landlab has a truly or at least largely implicit solution, it does _not_ seek to impose any internal restrictions on that timestep. As you rightly note, this runs the risk that the user can take inappropriately long timesteps. We agonised about this, but decided that in a very flexible environment like Landlab, adding arbitrary timestep restrictions on implicit methods would create more problems and confusion than it would solve. In such cases, the component documentation normally makes reference to this issue, and the user is encouraged to think about it for themselves – the "appropriate timestep" would be very much situation dependent, so we can't really legislate for it in a general way. We've added a sentence here to make explicit for the reader that this is indeed "a thing": "Note also that where components employ implicit solutions, there may be no internal limit to the timestep at all (e.g., Braun and Willett's (2013) Fastscape algorithms for stream power). In such cases, Landlab will make no check on the imposed timestep, and the user must ensure that the imposed dt is appropriate under the boundary and initial conditions that they are running. For instance, the Braun-Willett algorithm ceases to behave in a truly timestep-independent fashion under transient conditions, but in a way that still permits timesteps larger than would be imposed under an explicit Courant condition (for more details see their Appendix B). However, those authors did not propose an alternative scheme to limit the timestep in such cases, and consequently Landlab also does not. A user of this component is assumed to have read the component documentation and taken on board that this is potentially an issue, and to have taken steps to check that their output is behaving sensibly and is not highly sensitive to changes in the supplied timestep. We reiterate that it is ultimately the user's responsibility to check that the provided dt is appropriate to the modelling scenario in hand."

Page 18-19: nice illustration on the use of finite volume methods.

Page 22, line 31: what about maximum timesteps for implicit schemes? See comment above (Page 15-16) ν Now addressed explicitly in that section.

Page 24 line 22: I would get rid of the sentence starting with Figure 13. I also do not see the additional value of Fig 13 for this paper. ν We would prefer to retain the figure, as it neatly encapsulates a key functionality of Landlab that isn't well illustrated in any of the other figures.

Figures

Great figures in general.

Figure 1: Nice images but I find it a missed opportunity to illustrate the broad range of problems Landlab itself is capable to simulate. ν In principle, Landlab would be able to simulate any and all of these processes, given a motivated user. In most of these cases, moves are already afoot to add these functions into Landlab for a future release. . .

Not sure what the additional value of the code snippets is. Personally, I find the Landlab wiki page much more insightful in this perspective. ν While we definitely would want a user to look at the wiki, we wanted this paper to be able to stand alone, without forcing a reader to go online to get more resources. Hence we have included the code snippets in the text.

Testing:

I am not very familiar with python and I second the comments of referee 1 who did elaborate testing with a large user group. Nevertheless, I installed the software and executed some of the tutorials which I found very easy to understand and clearly documented. I had no single problem when executing the code. A very small suggestion would be to add a link to the 'coupled_params_storms.txt' file in the 'Getting to know the Landlab component library' tutorial on https://nbviewer.jupyter.org/github/landlab/tutorials/blob/master/component_tutorial/component_tutorial.ipynb, similar to the link given for the 'coupled_params.txt' file. ν Thanks for the recommendation. I'll add it as a ticket on Github.

Interactive comment on Earth Surf. Dynam. Discuss., doi:10.5194/esurf-2016-45, 2016.