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.

- 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.

- Input/output data structures. Compatibility with other tools available.

Well implemented and properly discussed in the paper

- 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. Nonetheless, 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)?

- 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?

- 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.

- 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 solutions 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

- 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.

- 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))).

- 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.

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).

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*?

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.

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)

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.

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.

Not sure what the additional value of the code snippets is. Personally, I find the Landlab wiki page much more insightful in this perspective.

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.