Exploring Surface Processes Using the Community Surface Dynamics Modeling System Modeling Tools

Jordan Adams, Postdoctoral Research Associate, The Institute of Arctic and Alpine Research, University of Colorado, Boulder, CO, jordan.adams@colorado.edu
Irina Overeem, Associate Professor, Department of Geological Sciences, University of Colorado, Boulder, CO, irina.overeem@colorado.edu
Eric Hutton, Chief Software Engineer, Community Surface Dynamics Modeling System, University of Colorado, Boulder, CO, eric.hutton@colorado.edu
Gregory Tucker, Professor, Department of Geological Sciences, University of Colorado, Boulder, CO, gtucker@colorado.edu
Albert Kettner, Research Scientist II, The Institute of Arctic and Alpine Research, University of Colorado, Boulder, CO, albert.kettner@gmail.com

Extended Abstract

Hydrological and sediment transport processes operate over a range of temporal and spatial scales. Flood events can cause catastrophic erosion rates over short timescales, reshaping floodplains and catchments in hours or over days. Over longer timeframes, from decades to millennia, the cumulative effect of these erosional events sculpt watershed morphologies, driving changes in drainage area, density and relief. The feedbacks between hydrologic events and sediment transport can shape areas as small as millimeter-scale hillslope rills or as large as continental-scale river basins.

The unpredictable or unobservable nature of flood events makes it difficult to study the interaction between hydrologic and sedimentological proccesses. The stochastic nature of floods make it challenging to predict when an event will occur and how to best measure erosional and depositional changes. Morphological responses to climate change occur on timescales too long to make meaningful observations. Moreover, at many relevant natural hazard scales, hydrology and sediment transport are entangled with ecosystem and human dynamics, complex interactions that are even less understood.

Improved understanding, and ultimately, improved resiliency in the face of hydrologically-driven Earth surface change, require computational models that bridge boundaries and link process mechanics. Many hydrological models exist and have a variety of uses: forecasting floods or water resource availability (e.g. WRF-Hydro, Gochis et al., 2018); channel and floodplain engineering (e.g. HEC-RAS, Brunner, 2016) or groundwater resource assessment (SWAT, USDA ARS Grassland Soil and Water Research Laboratory, 2018). As computational resources become more efficient, more researchers are adding numerical modeling skills to their repertoire. Yet, as more models are built, questions remain: can the surface processes community work together to share these ever-improving tools? Is there a way to standardize both existing and new modeling components so that they can be coupled flexibly and effectively?

The Community Surface Dynamics Modeling System (CSDMS) is a NSF-funded initiative that supports the open software efforts of the surface processes community. CSDMS sets modeling standards and protocols, hosts a Model Repository to distribute models and modeling tools, and provides cyberinfrastructure to an interdisciplinary set of community members. The CSDMS Repository contains over 200 tools and models that simulate lithosphere, hydrosphere, atmosphere or cryosphere dynamics. The goal of CSDMS is simple: to expedite scientific discovery and eliminate duplication of effort by sharing computational resources.

As part of these efforts, CSDMS has designed a new tool for hypothesis-driven modeling; the CSDMS Python Modeling Tool (PyMT) provides a unified framework that allows users to interactively run and couple numerical models written in a variety of programming languages. These coupled models can operate on disparate time and space scales, which are then resolved by PyMT. Principally, the PyMT is three things: (1) a collection of Earth-surface models wrapped with a common interface, (2) an extensible plug-in framework into which new models can be incorporated, and (3) tools for coupling models that operate on a variety of spatial grids and time steps.

Currently, the PyMT model collection consists of several dozen Earth-surface models that cover a variety of process domains that range from land, to coast, to ocean. Each of these models were contributed by CSDMS community members to the CSDMS Model Repository as standalone models. There was no initial intent for these models to be part of a larger framework. The heterogeneity of the collection is represented not just in the variety of programming languages but also by idiosyncratic user interfaces (e.g. model specific input and output file formats).

All models within the PyMT collection are wrapped in a single, unified, interface within the Python programming language. An overriding tenet of the PyMT is: *if you know how to use one pymt model, you know how to use all pymt models*. The PyMT model interface allows users to interactively run models within a Python kernel such that they can advance models through time while dynamically changing their state variables. This allows users to become model composers by orchestrating different model functionality within a script, while being able to leverage the power of the Python programming language and its powerful collection of third-party packages (e.g. numpy, scipy, matplotlib, xarray, dask).

The PyMT provides an extensible plugin framework that allows additional models to be easily incorporated into the PyMT framework. This allows new models, written by domain experts, to become PyMT models usable by a broad community in potentially novel ways. To be incorporated into PyMT, new models must be written to expose a Basic Model Interface (BMI). At its core, the BMI is simply a specification that defines the necessary functions a model must provide to make it coupleable. These functions control how a model is initialized, updates through time, as well as how a model provides its output variables or ingests externally provided input variables. The CSDMS modeling stack additionally provides tools for automatically wrapping models written in several programming languages (currently C, C++, Fortran, and Python) into PyMT. These four languages cover a significant majority of the models in the CSDMS Model Repository.

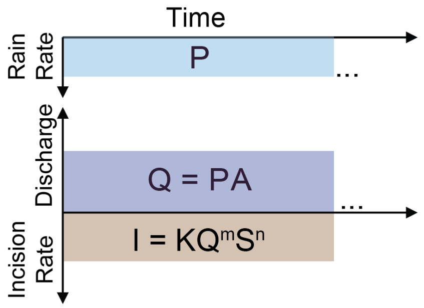
The PyMT contains a collection of tools useful for model coupling (either model-to-model or model-to-data coupling). As mentioned previously, the CSDMS Model Repository is a heterogeneous collection of models that were not necessarily written with the intent of coupling to other models. As such, a significant obstacle when coupling models is transferring values from one model's solution grid to another. Included with the PyMT are a set of grid mappers, based on the Earth System Modeling Framework (ESMF) grid mapping library. This allows for efficient mapping of values between large grids. Other coupling tools include:

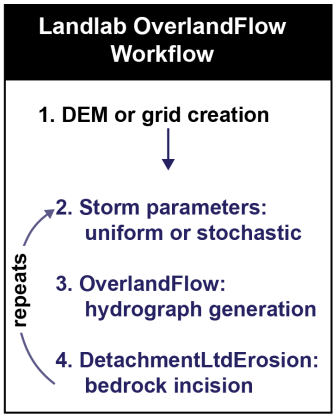
- Unit conversion utilities, which conform to the cfunits conventions, when models provide the same values but with different units,
- Time interpolators that estimate state variables between model timesteps,
- Output writers that write model output to standardized netCDF files that conform to the UGRID/SGRID specifications

PyMT is designed to expedite the process of exploring ideas, testing hypotheses, and comparing models with data, and make Earth-surface process models more accessible.

For proof of concept, we present an example of a coupled hydrodynamic and bedrock incision model in PyMT. This work uses two components from the Landlab model, OverlandFlow and DetachmentLtdErosion, as they are implemented in PyMT (Adams et. al, 2017; Hobley et al., 2017). The OverlandFlow component was originally developed to bridge the gap between fully hydrodynamic models used to model single hydrograph events and simplified 'steady-state' hydrology components used in long-term fluvial geomorphology models. Figure 1 illustrates the difference between these two model types: 'steady-state' models often simplify rainfall and runoff into steady, constant values that drive steady, constant incision rates (Figure 1a), while non-steady models can take rainfall stochasticity into account and drive individual event hydrographs and changing incision rates through time (Figure 1b). As computational efficiency and speed have improved over the last decade, more efforts have been made to bring hydrodynamics into models of landscape evolution. The Landlab OverlandFlow model is an open-source tool designed to achieve that goal.

(a) Steady-state hydrology model





(b) Nonsteady hydrology model

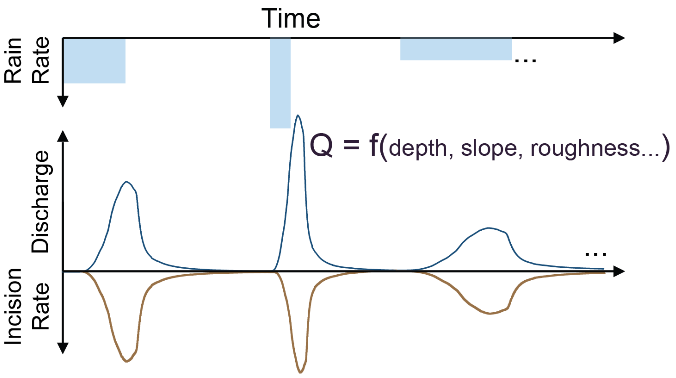


Figure 1. Comparison of steady (**a**) and nonsteady (**b**) hydrology models. The latter case illustrates how the OverlandFlow model is implemented. A sample OverlandFlow workflow is also shown.

In this presentation, we run several test cases in PyMT to illustrate landscape sensitivity to rainfall parameters, hydrograph shape and basin orientation. These results are compared against traditional steady-state model results. Landscapes eroded and evolved using nonsteady methods are characterized by greater relief and increased channel concavities when compared to steady results, suggesting that hydrodynamics should be considered when studying the impact of bedrock river incision on topographic evolution over long timescales.

The implementation of OverlandFlow and DetachmentLtdErosion is just one example of coupled hydrology-sedimentology modeling available through PyMT. This presentation will provide detailed background on how models can be brought into the PyMT framework, how PyMT resolves grid and temporal differences across models, the existing hydrologic and sedimentologic tools in PyMT, and examples of model output from the OverlandFlow and DetachmentLtdErosion models.

References

Adams, J.M., Gasparini, N.M., Hobley, D.E., Tucker, G.E., Hutton, E., Nudurupati, S.S. and Istanbulluoglu, E., 2017. The Landlab v1.0 OverlandFlow component: a Python tool for computing shallow-water flow across watersheds. Geoscientific Model Development, 10(4).

Brunner, G. W. (2016). HEC-RAS river analysis system 2D modeling user's manual. US Army Corps of Engineers—Hydrologic Engineering Center, 1-171.

Gochis, D.J., M. Barlage, A. Dugger, K. FitzGerald, L. Karsten, M. McAllister, J. McCreight, J. Mills, A. RafieeiNasab, L. Read, K. Sampson, D. Yates, W. Yu, 2018. The WRF-Hydro modeling system technical description, (Version 5.0). NCAR Technical Note. 107 pages. DOI:10.5065/D6J38RBJ

Hobley, D.E., Adams, J.M., Nudurupati, S.S., Hutton, E.W., Gasparini, N.M., Istanbulluoglu, E. and Tucker, G.E., 2017. Creative computing with Landlab: an open-source toolkit for building, coupling, and exploring two-dimensional numerical models of Earth-surface dynamics. Earth Surface Dynamics, 5(1), p.21.

USDA ARS Grassland Soil and Water Research Laboratory; Texas A&M AgriLife Research, 2018. SWAT - Soil and Water Assessment Tool. USDA Agricultural Research Service; Texas A&M AgriLife Research. https://data.nal.usda.gov/dataset/swat-soil-and-water-assessment-tool