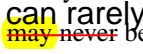
The Landlab OverlandFlow component: a Python library for computing shallow-water flow across watersheds

Jordan M. Adams[1], Nicole M. Gasparini[1], Daniel E. J. Hobley[2], Gregory E. Tucker[3,4], Eric W. H. Hutton[5], Sai S. Nudurupati[6], and Erkan Istanbulluoglu[6]

[1] Department of Earth and Environmental Sciences, Tulane University, New Orleans, USA
[2] School of Earth and Ocean Sciences, Cardiff University, Cardiff, UK
[3] Cooperative Institute for Research in Environmental Sciences (CIRES), University of Colorado, Boulder, USA
[4] Department of Geological Sciences, University of Colorado, Boulder, USA
[5] Community Surface Dynamics Modeling System (CSDMS), University of Colorado, Boulder, USA
[6] Department of Civil and Environmental Engineering, University of Washington, Seattle, USA

Correspondence to: J. M. Adams (jadams15@tulane.edu)

Abstract. Hydrologic models and modeling components are used in a wide range of applications. Rainfall-runoff models are used to investigate the evolution of hydrologic variables, such as soil moisture and surface water discharge, throughout one or more rainfall events. Longer-term landscape evolution models also include aspects of hydrology, albeit in a highly simplified manner, in order to approximate how flowing water shapes landscapes. Here we illustrate how the OverlandFlow hydrologic
5 component contained within Landlab can be applied as either a short-term rainfall-runoff model or a longer-term landscape evolution model. Landlab is a Python-language library that includes tools and process components that can be used to create models of Earth-surface dynamics over a range of temporal and spatial scales. The Landlab OverlandFlow component is based on a simplified inertial approximation of the shallow water equations, following the solution of de Almeida et al. (2012). This explicit two-dimensional hydrodynamic algorithm propagates a flood wave across a terrain, and water discharge and flow
10 depth are calculated at all locations within a structured (raster) grid. Examples of flow routing on both real and synthetic landscapes are shown. Hydrographs from a single storm at multiple locations in the Spring Creek watershed, Colorado, USA, are illustrated, along with maps of water depth and shear stress applied on the surface by the flowing water. Flow routing on two different synthetic watersheds illustrates how network organization impacts hydrograph shape. The OverlandFlow component is also coupled with the Landlab DetachmentLtdErosion component to illustrate how the nonsteady flow routing
15 regime impacts incision across a watershed. The hydrograph and incision results are compared to simulations driven by steady-state runoff, or discharge equal to the product of drainage area and rainfall rate, which is the norm in landscape evolution modeling. Results from the coupled hydrologic and incision model indicate that runoff dynamics can impact landscape relief and channel concavity. Example code is provided that demonstrates how to use the OverlandFlow component and couple it with other components to create a model.

1 Introduction

Numerical models of overland flow have a variety of applications. Examples include mapping rb in flooding events (e.g. Dutta
et al., 2000; Horritt and Bates, 2002), modeling hydrogeomorphologic processes in post-wildfire landscapes (e.g. Beeson et al.,
2001; Rengers et al., 2016) and the interactions between surface and subsurface water by way of soil infiltration (e.g Esteves
5   et al., 2000; Kollet and Maxwell, 2006). Another possible application is the study of erosion and sedimentation driven by
overland flow, and how these processes can shape long-term landscape evolution.

Models of landscape evolution all have the same fundamental structure: all use numerical methods to model flow or transport
of water and sediment across a representative mesh that is tessellated into discrete elements (e.g. Coulthard, 2001; Willgoose,
2005; Tucker and Hancock, 2010; Chen et al., 2014, and references therein). To some degree, all landscape evolution software
10   packages model interactions between hydrology and geomorphology, but the complexity of the hydrologic component varies
(e.g. Willgoose et al., 1991; Tucker and Slingerland, 1994; Willgoose, 1994; Braun and Sambridge, 1997; Tucker et al., 2001;
Coulthard, 2001; Coulthard et al., 2002; Hancock et al., 2015). The representation of hydrological processes in landscape
evolution models is often simplified. In particular, overland flow processes are simplified as there are complex ties when
solving the shallow water equations. Most models assume steady-state water discharge, where surface water flux is modeled
15   at each location as a function of drainage area and precipitation rate, and flow is only present in the system for the duration
of a precipitation event. Flow routing algorithms can be simplified as well, limiting flow to travel along the steepest descent
out of a given location (Tarboton, 1997; Tucker et al., 2001; Tucker and Hancock, 2010). These simplifications are often
made for two reasons: there can be significant differences between hydrologic time steps for individual flood and storm events
(seconds to days) and geomorphic time steps of rock uplift and landscape evolution (years to millennia) that may be complex
20   to resolve; additionally, computational power may be a limiting factor, and these simplifying assumptions may speed up the
model processing time.

Whereas many geomorphic landscape evolution models generalize hydrology using steady-state assumptions, there are many
hydrologic and flood inundation models that route a storm hydrograph (changing discharge through time), capturing the spatial
and temporal variability of water discharge across a modeled landscape (e.g. Bates and De Roo, 2000; Ogden et al., 2002;
25   Downer and Ogden, 2004; Ivanov et al., 2004; Hunter et al., 2007; Moradkhani and Sorooshian, 2009; Bates et al., 2010;
de Almeida et al., 2012; Devi et al., 2015, and references therein). Some of these hydrologic models have been paired with
erosional models at the watershed scale (e.g. Aksoy and Kavvas, 2005; Francipane et al., 2012; Coulthard et al., 2013; Kim
et al., 2013, and references therein). However, there are a limited number of studies that integrate a physically-based hydrologic
model into a landscape evolution modeling framework, due to the challenges previously discussed.

30   The assumption of steady-state runoff in landscape evolution models is often not reasonable. For example, steady-state
hydrologic conditions can rarely ~~may never~~ be achieved in larger catchments with long flow paths, or in landscapes dominated by short-
duration precipitation events. Adding hydrologic variability in landscape evolution models has been shown to impact watershed
morphology. Previous work coupling spatially variable rainfall models with steady-state discharge in landscape evolution
models has shown controls on landform morphology, including relief and drainage network organization (e.g. Anders et al.,

2008; Colberg and Anders, 2014; Huang and Niemann, 2014; Han et al., 2015). Similarly, introducing discharge variability into landscape evolution models has implications for incision rates, channel profile form and steepness in modeled landscapes (e.g. Tucker and Bras, 2000; Lague et al., 2005; Molnar et al., 2006; DiBiase and Whipple, 2011). In contrast to these studies, Coulthard et al. (2013) integrated a semi-implicit hydrodynamic model into the CAESAR landscape evolution model and noted

5 reduced sediment yields on decadal time scales of landscape evolution when using nonsteady hydrology. In another approach, Sólyom and Tucker (2004) estimated nonsteady peak discharge as a function of the storm duration, rainfall rate and the longest flow length in a network. Incision rates were estimated using those peak discharge values. Their findings demonstrated that landscapes evolved with nonsteady hydrology were characterized by decreased valley densities, reduced channel concavities and increased relief when compared to landscapes evolved using steady-state hydrology.

10 To represent and investigate the role of nonsteady flow routing on landform evolution, a hydrodynamic model has been incorporated into the new Landlab modeling toolkit. In this paper, we describe the fundamentals of the Landlab modeling framework, as well as the theoretical background of the Landlab OverlandFlow component, based on a two-dimensional flood inundation model (LISFLOOD-FP: Bates and De Roo, 2000; Bates et al., 2010; de Almeida et al., 2012; de Almeida and Bates, 2013). This description of the new OverlandFlow component includes information on how to set up a model domain using a

15 digital elevation model, how to handle boundary conditions and model data, and the validation against a known analytical solution. The OverlandFlow component is then used to route nonsteady flow on one real landscape and two synthetic watersheds. The OverlandFlow component can be used to explore implications of a nonsteady hydrology method on short-term landscape evolution. Model output demonstrates that the OverlandFlow component is sensitive to both catchment characteristics and precipitation inputs. Output hydrographs can be flashier or broader depending on changes in these parameters. Finally, the variable

20 discharge from the OverlandFlow component is coupled to a detachment-limited erosion component (DetachmentLtdErosion) to explore the feedbacks between hydrograph shape and short-term erosion patterns throughout the landscape.

2 Landlab modeling framework

Landlab is a Python-language, open-source modeling framework, developed as a highly flexible and interdisciplinary library of tools used to address scientific problems in Earth-surface dynamics (Adams et al., 2014; Tucker et al., 2016; Hobley et al.,

25 in review). The utilities in Landlab allow users to build two-dimensional numerical models (Fig. 1). This includes a gridding engine that creates structured or unstructured grids, a set of pre-built components that implement code representing Earth surface or near-surface processes, and structures that handle data creation, management and sharing across the different process components. A diverse group of processes, such as uniform precipitation, detachment- and transport-limited sediment transport, crustal flexure, soil moisture, vegetation dynamics, and overland flow, are available in the Landlab library as process

30 components. The Landlab architecture allows for a "plug-and-play" style of model development, where process components can be coupled together. Coupled components share a grid instance and methods, and can operate on the data attached to the grid.

2.1 RasterModelGrid library

Landlab offers several different grid types. However, because the core algorithm in the OverlandFlow component can only be applied to rectangular grids, only the RasterModelGrid (structured grid) class is described here. The RasterModelGrid class can build both square ($\Delta x = \Delta y$) and rectilinear ($\Delta x \neq \Delta y$) grids. Each grid type in Landlab is composed of the same topological

5   elements: nodes, which are points in (x, y) space; cells, a polygon with area $\Delta x \Delta y$ surrounding all non-perimeter or interior nodes; and links, ordered line segments which connect neighboring pairs of nodes and store directionality (Fig. 2). In the RasterModelGrid library, each node has four link neighbors, each oriented in a cardinal direction. Each node has two "inlinks", connecting a given node to its south and west neighbors, and two "outlinks", connecting to the neighbors in the north and east. The terms "inlinks" and "outlinks" are for topological reference only, as the direction of fluxes in a typical Landlab component

10   are calculated based on link gradients.

Model data can be stored on these grid elements using Landlab data fields. The data fields are NumPy array structures that contain data associated with a given grid element. To store and access data on these fields, data is assigned using a string keyword, and is accessed using Python's mutable dictionary data structure. Data is attached to the grid instance using these fields, and can be accessed using the string name keyword and updated by multiple Landlab components. For example, a field of val-

15   ues representing water depth at a grid node can be accessed using the following syntax: *grid.at_node['surface_water__depth']*, where *grid* is the grid instance. Most Landlab names follow a simplified version of the naming conventions of the Community Surface Dynamics Modeling System (CSDMS), a set of standard names used by several models within the Earth science community (Peckham, 2014).

Model boundary conditions are set within a Landlab grid object. Boundary conditions are set on nodes and links (Fig. 3).

20   Node boundary statuses can be set to either *boundary* or *core*. If a node is set to boundary, it can be further defined as an open, fixed gradient, or closed (no flux) boundary. In all RasterModelGrid instances, default boundary conditions are set as follows: perimeter nodes are open boundary nodes and interior nodes are core nodes. Boundary conditions can also be applied to interior nodes (e.g. NODATA values in the interior of a digital elevation model can be set as closed boundaries). There are three link boundary statuses: active, inactive and fixed. Link boundary status is tied to the neighboring nodes. Once boundary conditions

25   are set on the nodes, link boundary conditions are automatically updated. Active links occur where fluxes are calculated, and are found in two cases: (1) between two core nodes or (2) between one core node and one open boundary node. Fixed links can be assigned a fixed value that can be set or updated during the model run and are located between a fixed gradient node and a core node. Fluxes are not calculated on inactive links, which occur in two cases: (1) between a closed boundary and a core node or (2) between any pair of boundary nodes of any type (Fig. 3). Core nodes and active links make up the computational

30   domain of a Landlab model.

3 Component equations

3.1 deAlmeida OverlandFlow component

Solving explicit hydraulic formulations can be computationally challenging. For example, the one-dimensional shallow water equation includes four terms:

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A_{xs}}\right) + gA_{xs}\frac{\partial (h+z)}{\partial x} + \frac{gn^2|Q|Q}{R^{4/3}A_{xs}} = 0 \tag{1}$$

where Q is water discharge $[L^3T^{-1}]$; t is time $[T]$; x is the location in space; A_{xs} is cross-sectional area of the channel; g is gravitational acceleration $[LT^{-2}]$; h is water depth $[L]$; z is the bed elevation $[L]$; n is the Manning's friction coefficient $[L^{-1/3}T]$ and R is the hydraulic radius $[L]$. These terms represent, from left to right, local acceleration, advection, fluid pressure and friction slope. To enhance stability, however, many solutions of the shallow water equations include numerical approximations that neglect terms from this solution. The simplest approximation, the kinematic wave model, neglects the local acceleration, advection and pressure terms. A more complex approximation, the diffusive wave model only neglects the local acceleration and advection terms (Kazezyılmaz-Alhan and Medina Jr, 2007).

The Landlab OverlandFlow component is based on the two-dimensional hydrodynamic algorithm developed for the LISFLOOD-FP model, and similar to the diffusive approximation, assumes a negligible contribution from the advection term of the shallow water equations (Bates et al., 2010; de Almeida et al., 2012). Additionally, this solution assumes a rectangular channel structure and constant flow width, impacting the pressure and friction terms (A_{xs} and R) in Eq. (1). This formulation allows for a larger maximum time step than the more common diffusive approximation, enhancing the computational efficiency of the OverlandFlow component (Bates et al., 2010). de Almeida et al. (2012) further stabilized this algorithm by introducing a diffusive term into LISFLOOD-FP, updating the Bates et al. (2010) algorithm to work on lower friction surfaces without sacrificing computational speed.

To start the model, a stable time step is calculated. If a too-large time step is used, areas of low slope are prone to wave oscillations, leading to a spatial 'checkerboard' pattern of water depths. Too-small time steps can have significant impacts on the computational performance of a model. To maximize the trade-off between computational efficiency and stability of the de Almeida et al. (2012) solution, an adaptive time step (following Hunter et al., 2005) is used to keep the Courant-Freidrichs-Levy (CFL) condition valid:

$$\Delta t_{max} = \alpha \frac{\Delta x}{\sqrt{gh_{max}}} \tag{2}$$

where Δt_{max} is the maximum time step that adheres to the CFL condition; α is a dimensionless stability coefficient less than 0.7; Δx is the grid resolution $[L]$; and $\sqrt{gh_{max}}$, the characteristic velocity of a shallow water wave, or the wave celerity $[LT^{-1}]$, calculated using h_{max}, the maximum depth of water in the modeling domain $[L]$. When the OverlandFlow component is initialized, a thin film of water is set at all grid nodes to keep Eq. (2) valid. Parameters and variables are defined in Tables (1), and (2).

To calculate water discharge, de Almeida et al. (2012) derived an algorithm using the one-dimensional Saint-Venant or shallow water equations which simulate a flood wave propagating across gridded terrain (for full derivation see deAlmeida et al., 2012). The explicit solution follows the form:

$$q_x^{t+\Delta t} = \frac{[\theta q_x^t + \frac{1-\theta}{2}(q_{(x-1)}^t + q_{(x+1)}^t)] - g h_{f(x)} \Delta t S_{w(x)}}{1 + g \Delta t n^2 |q_x^t| / h_f^{7/3}} \tag{3}$$

5   where q is water discharge per unit width $[L^2 T^{-1}]$, calculated on links, here given superscript t for the current time step and subscript x describing the location of links in space (Fig. 4). θ is a weighting factor between 0 and 1, given a default value of 0.8, but can be tuned by the user. Setting θ to 1 returns the semi-implicit solution of Bates et al. (2010), that is, removing the diffusive effects implemented by de Almeida et al. (2012). g is gravitational acceleration $[LT^{-2}]$; h_f is the local maximum water surface elevation at a given time $[L]$; Δt is the adaptive time step $[T]$ (Eq.2); S_w is the dimensionless water

10   surface slope; and n is the Manning's friction coefficient $[L^{-1/3}T]$ (Tables 1 and 2). Equation (3) is calculated as two one-dimensional solutions in a D4 (four-direction) scheme: first calculated in the east-west direction (in the x direction) and then in the north-south direction (replacing x with y in Eq. 3).

Water depth is calculated on nodes, and updated at each time step as a function of the surrounding volumetric water fluxes on both horizontal and vertical links:

15   $$\frac{\Delta h}{\Delta t} = \frac{Q_{h(in)} - Q_{h(out)}}{\Delta x \Delta y} \tag{4}$$

where $Q_{h(in)}$ $[L^3 T^{-1}]$ are the summed water discharges moving into a given node and $Q_{h(out)}$ are summed water discharges moving out of a given node, following Fig. (2). Directionality of discharge is determined not by the orientation of "inlinks" or "outlinks", but instead, flow directions are determined by the gradient of each link. In this method, water mass is conserved, as the flow moving out of a node is balanced by the flow moving into the nearest node neighbor.

20   ### 3.1.1   Steep environment stability criteria

The de Almeida et al. (2012) equation is designed for urban flooding events and is most stable in flat environments. To adjust this component to work in steep mountain catchments, extra stability criteria were added to keep the simulation numerically stable. Similar criterion were implemented in the CAESAR-Lisflood model (Coulthard et al., 2013). This method reduces the calculated flow discharge as needed to keep flow regime critical to subcritical using the Froude number (Eq. 5), where

25   subcritical flow is defined as Fr ≤ 1.0. The Froude number is calculated as a function of wave velocity (u, calculated as $\frac{q}{h_f}$ on all links) and wave celerity ($\sqrt{gh_f}$):

$$Fr = \frac{u}{\sqrt{gh_f}} \tag{5}$$

If the $steep slopes$ flag is set when initializing OverlandFlow, restrictions are imposed to keep flow conditions critical to subcritical, a reasonable assumption for steep, mountain catchments (Grant, 1997). Specifically, if the water velocity calculated

30   by the component drives the Froude number > 1.0, water velocity is reduced to a value that maintains a Froude number ≤ 1.0 for that given time step.

3.2 DetachmentLtdErosion component

To illustrate the flexibility of the OverlandFlow component, we present an example in Section 7, in which water discharges driven by overland flow are coupled with surface erosion. Specifically, we explore a case where incision rate is solved explicitly, and depends on local water discharge and water surface gradient (e.g. Howard, 1994; Whipple and Tucker, 1999, 2002;
5   Pelletier, 2004). This equation follows the form:

$$I = KQ^{m_{sp}}(S_{w_{max}})^{n_{sp}} - \beta \tag{6}$$

where I is the local incision rate $[LT^{-1}]$; K is a dimensional erodibility coefficient units depend on the positive, dimensionless stream power coefficients m_{sp} and n_{sp}; Q is total water discharge on a node at a given time step $[L^3 T^{-1}]$; $S_{w_{max}}$ is the local maximum water surface slope, which is dimensionless, and β is the optional threshold, below which no change in bed
10   elevation is permitted $[LT^{-1}]$ (Tables 1 and 2). β is commonly interpreted as an entrainment threshold for bedload at rest on the bed in between erosional events (e.g. Attal et al., 2011). By default, m_{sp} and n_{sp} have set values of $m_{sp} = 0.5$ and $n_{sp} = 1.0$ that can be adjusted by the model user.

This solution allows for only the local detachment of material and assumes that transport rate is much larger than sediment supply rate. Therefore, no deposition is considered here. This erosion formulation is implemented with the Landlab Detach-
15   mentLtdErosion component. A threshold can be applied, under which no erosion is able to occur. For simplicity, no threshold is assumed here.

4   OverlandFlow model implementation in Landlab

To use the coupled Landlab OverlandFlow and DetachmentLtdErosion model, the user interacts with a driver file (Fig. 1). A simple Landlab driver file can run a model using fewer than 20 lines of code (Algorithm 1). There are four parts to running
20   the coupled OverlandFlow-DetachmentLtdErosion model: (1) creating a domain using the RasterModelGrid, either explicitly or using a digital elevation model (DEM) in the ArcGIS ASCII format; (2) setting boundary conditions on the domain; (3) initializing the components; and (4) coupling them using the Landlab field data structures.

4.1   Initializing a grid: user-defined or DEM

To set up a grid instance, the user can create a rectangular grid by passing the number of rows, number of columns and grid
25   resolution (Δx) as keywords to the RasterModelGrid object. This can be accomplished in one line of code:
$grid = RasterModelGrid((number\_of\_node\_rows, \ number\_of\_node\_columns), \ \Delta x)$. In this method, only an empty instance of the grid is created, so elevation data must be assigned to grid nodes by the user.

An alternative method is to read in gridded terrain data from other file types. The original intent of Bates et al. (2010) was to develop a new flood inundation algorithm that can work easily with the growing availability of terrain data collected by
30   satellite, airborne, or terrestrial sensors. Landlab's input and output utilities simplify this process by including functionality to

read in data from an ASCII file in the Esri ArcGIS format (Algorithm 1, Line 3). In this method, elevation data is read in and automatically assigned to a Landlab data field called $topographic\_elevation$, set using the *name* keyword.

4.2 Boundary condition handling

Node boundary conditions are set throughout the grid in a Landlab OverlandFlow model to delineate the modeling domain
5 (Algorithm 1, Line 4). For flow to move out of a watershed or system, an open boundary must be set at the outlet. If the node location of the outlet is unknown, there is a utility within the grid ($set\_watershed\_boundary\_condition$, Algorithm 1, Line 4) that will find the outlet and set it as an open boundary, in addition to setting all NODATA nodes to closed boundaries across the DEM or model domain.

The de Almeida et al. (2012) equation uses neighboring link values when calculating water discharge (Fig. 4). By default,
10 the edge of the watershed links are set to inactive status, and are assigned a value of 0, simulating no input from outside of the watershed for the simulation. If the user wants to simulate an input discharge on these links, an alternative method is the $set\_nodata\_nodes\_to\_fixed\_gradient$ method. If this method is called, the user can manually update discharge values on links with FIXED_LINK boundary status outside of the OverlandFlow class. Fixed links are accessed through their IDs using the RasterModelGrid class ($grid.fixed\_links$). In this method, the user can set a discharge value per unit width $[L^2 T^{-1}]$ on
15 all fixed links. This method is advised if the user has a known input discharge they want to force at the watershed or domain edge.

4.3 Initialize OverlandFlow and DetachmentLtdErosion

Landlab components have a standard initialization signature and take the grid instance as the first keyword (Algorithm 1, Lines 6-8). Any default parameters are also in the component signature and can be updated when the component is called. These
20 parameters can be adjusted according to the physical nature of the landscape being tested. For the OverlandFlow component, Eq. (3) parameters Manning's n and discharge weighting factor θ can be adjusted. A steady, uniform precipitation rate can also be passed as a system input using the $rainfall\_intensity$ parameter (Algorithm 1, Line 7). Additionally, a stability criterion flag for steep catchments can be set ($steep\_slopes$ = TRUE, as described in Section 3.1.1.). In the DetachmentLtdErosion component, stream power exponents m_{sp} and n_{sp} and erodibility parameter K are also set by passing arguments to the component
25 on instantiation.

4.4 Coupling using Landlab fields

To couple the OverlandFlow and DetachmentLtdErosion components, values for water discharge (Q_h), surface water slope (S_w) and topographic elevation (z) are shared as data fields through the RasterModelGrid instance (e.g. Algorithm 1, Lines 14-15). At each time step, the water discharge and surface water slope fields are updated by the OverlandFlow component (Eq. 3).
30 These new values are used to calculate an incision rate in the DetachmentLtdErosion component (Eq. 6). At each grid location,

topographic elevation (z) is redu according to the incision rate. Changes in topographic slope caused by erosion throughout the landscape will drive changes in surface water slope ($S_{w_{max}}$) in the next iteration of the OverlandFlow component.

5 Analytical solution

To validate the OverlandFlow component, we compared model output against an analytical solution for wave propagation on a flat surface, following Hunter et al. (2005). This test case propagates a wave over a flat horizontal surface (assuming a slope of 0), given a uniform friction coefficient (n) and constant, single-direction velocity (u) (For full derivation see: Hunter et al., 2005; Bates et al., 2010; de Almeida et al., 2012), This solution is written as:

$$h(x,t) = \left[-\frac{7}{3}\left(n^2 u^2 \{x - ut\} \right) \right]^{\frac{3}{7}} \tag{7}$$

Solving for the leftmost boundary of the modeling domain ($x = 0$) gives:

$$h(0,t) = \left(\frac{7}{3} n^2 u^3 t \right)^{\frac{3}{7}} \tag{8}$$

To test the Landlab OverlandFlow component, a RasterModelGrid instance with dimensions of 32 rows by 240 columns was initialized with a resolution of $\Delta x = \Delta y = 25$ m. The water depth boundary condition through time (Eq. 8) is applied to left edge of the domain, while the top, right and bottom edges of the grid are set to CLOSED_BOUNDARY status to keep flow moving uniformly to the east and contained within the computational domain.

Two analytical solution test cases were run on the domain. The first is a low friction test ($n = 0.01$ sm$^{-1/3}$, $u = 0.4$ ms^{-1}, Fig. 5a,c) following the solution of Bates et al. (2010); de Almeida et al. (2012, Fig. 2). In the second the friction value is increased by an order of magnitude to closer approximate Manning's n values found in natural landscapes, while velocity was unchanged ($n = 0.1$ sm$^{-1/3}$, $u = 0.4$ ms^{-1}, Fig. 5b,d). To mirror previous tests using the LISFLOOD-FP model, Fig. (5) shows the water depth of wave fronts at three model times: $t = 2700$, 5400 and 9000 s. Each dashed line represents a changing theta value in Eq. (3), with $\theta = 1.0$ representing the semi-implicit solution from Bates et al. (2010). In all velocity-roughness conditions, the wave fronts predicted by the Landlab OverlandFlow component correlate well with the analytical solution defined using Eq. (8). In the low friction case ($n = 0.01$, Fig. 5a,c), the wave speed produced using the Landlab OverlandFlow is slower than the predicted wave front speed. Increasing surface roughness ($n = 0.1$, Fig. 5b,d), leads to the predicted wave front overestimating the analytical solution. Overall, the close approximation of the modeled solutions to known analytical solutions, across a wide range of roughness values, demonstrate the efficacy of the Landlab OverlandFlow component.

6 Application: Modeling OverlandFlow in a real landscape

6.1 Background

Most rainfall-runoff models are applied over real landscapes to simulate hydrologic events. Most rainfall-runoff models can be classified as either lumped or distributed. Lumped rainfall-runoff models represent watersheds as characteristic subareas

or subbasins, and do not account for spatial variability in subbasin parameters. These models assume that average variables and parameters adequately capture the processes being observed (Moradkhani and Sorooshian, 2009; Beven, 2011; Devi et al., 2015). Some examples of lumped hydrologic models include: Hydrological Simulation Program–Fortran (HSPF Donigan et al., 1984) and HEC-HMS (Scharffenberg and Fleming, 2006).

5   Alternatively, distributed rainfall-runoff model domains are broken into smaller, discrete elements or grid cells. Distributed models allow for spatial variability in model parameters or state variables. Existing distributed models include TOPMODEL, (Beven and Kirkby, 1979), KINEROS2 (Woolhiser et al., 1990), GSSHA (Downer and Ogden, 2004), and tRibs (Ivanov et al., 2004). Like these models, the Landlab OverlandFlow component can be used as a distributed model, routing rainfall across a real landscape DEM and estimating runoff for every point within a discrete RasterModelGrid instance. Changing discharge

10   values are calculated at every point in the watershed. Updated water depths, driven by changing discharge, can be used to calculate shear stress following the depth-slope product:

$$\tau = \rho g h S_w \tag{9}$$

Equation 9 calculates the bed shear stress τ $[ML^{-1}T^{-2}]$ as a function of fluid density (ρ) $[ML^{-3}]$, g, gravity; h, water depth; and S_w surface water slope. Shear stress exerted on the bed can be used to estimate sediment transport driven by flowing

15   water throughout the domain.

Here we illustrate a single storm routed across a DEM. In addition to water discharge, water depth and bed shear stress are calculated by the model, and plotted at the peak of the outlet storm hydrograph. This implementation of the OverlandFlow component illustrates how hydrologists can use Landlab as a distributed rainfall-runoff model to predict the hydrologic and sedimentologic impact of a single storm on a real landscape. These results demonstrate how the model can be used to predict

20   flooding and erosion events.

6.2   Methods: domain and parameterization

To apply the OverlandFlow component as a rainfall-runoff model, a DEM can be read into Landlab and converted easily into a RasterModelGrid instance. For this example, the Spring Creek watershed is used. Spring Creek is a steep, 27 km^2 watershed, located within Pike National Forest in central Colorado, USA (Fig. 6a). This LiDAR-derived DEM has square cells

25   with a resolution of $\Delta x = \Delta y = 30$ m (DEM data: Tucker, 2010). Using the $set\_watershed\_boundary\_condition$ utility, all NODATA nodes in the DEM are set to CLOSED_BOUNDARY status (Algorithm 1, Line 4). This method identifies the lowest elevation point within the watershed, the outlet, and sets it to an OPEN_BOUNDARY.

The DEM was pre-processed and pit-filled in a D4 routing scheme, where all nodes have at least one downstream neighbor in one of the four cardinal directions (Algorithm 1, Lines 8-9). This step ensures all flow can be removed from the domain. If

30   this step were to be skipped, flow may pond in "lakes" or "pits" in the domain, where flow cannot travel out of a given node location until the water surface elevation of the lake exceeds the bed elevation of the D4 neighboring nodes.

To initiate flow across the domain, a single storm was routed across the watershed. The 'base storm' (Table 3) was used as an example, with an effective rainfall rate of 5 mm hr^{-1} and a duration of 2 hr. For this storm, hydrographs were recorded at three

points throughout the watershed. At the peak of the outlet hydrograph, water depth at every location in the watershed was used to calculate the shear stress, which can be used to make interpretations about the transport of sediment across the watershed as a result of the storm.

6.3 Results and implication

5 In order to illustrate the downstream movement of the flood wave, hydrographs were plotted at three locations within the watershed for the duration of the flow event. The three hydrographs correspond to the three starred locations on the watershed DEM in Fig. 6a: at the outlet (black line, Fig. 6b), the approximate midpoint of the main channel (violet line, Fig. 6b) and an upstream location in the main channel (lavender line, Fig. 6b). In these hydrographs, both peak discharge and time to peak increase as the sampling site nears the outlet (moving from lighter to darker color). This pattern is expected as water

10 accumulates in the main channel from the tributaries. Upstream points have less contributing area, and so less water passes through those locations. The peak in the upstream-most hydrograph occurs first, as the flood wave passes through that location before propagating downstream. Downstream, the outlet has the largest contributing area in the watershed. Because the flow path is longest from the upstream reaches to the outlet, the time to peak is greater than upstream hydrographs. The outlet hydrograph is driven by contributing flow from all upstream points, increasing the peak discharge value.

15 Water depths are variable at each point throughout the model run, changing as a function of discharge inputs, outputs and effective rainfall rate at each time step (Eq. 4). Water depth values can be mapped across the domain at discrete time steps. In this example, water depth was plotted at the peak of the outlet hydrograph (Fig. 6c). At this peak, there is still water in the domain ready to flow out as part of the rising limb. These water depths can be used to calculate shear stress (following Eq. 9), also plotted at the peak of the outlet hydrograph (Fig. 6d). Shear stress (τ) values can be used to interpret the size of particles

20 that can be entrained and transported by channelized flow. Greater τ values correspond to areas with greater water depths (e.g. channels), where more sediment transport would be expected in high flow conditions.

In this example, we illustrate the flow of hydrographs across a real landscape and the resulting shear stress values. These results can be used to explore the processes controlling overland flow in a gauged landscape. Shear stress values can be used to estimate sediment transport rates, and make interpretations about spatial patterns of erosion and deposition, as well as

25 total sediment yields for particular storm events. These values could be calibrated in order to explore landscape sensitivity to rainfall-runoff events.

7 Application: Long-term fluvial erosion in Landlab

7.1 Background

Most landscape evolution models simplify hydrology by assuming steady-state, calculated as:

30 $$Q_{ss} = PA \tag{10}$$

where Q_{ss} is the steady-state water discharge $[L^3 T^{-1}]$, P is a constant effective precipitation or runoff rate $[LT^{-1}]$ and A is drainage area $[L^2]$. Discharge is steady for the duration of a precipitation event and stops when precipitation ends. Figure 7 compares this steady-state hydrology assumption against the physically-based nonsteady method at one location in a watershed. The effective rainfall rate P is the same rate and has the same duration for both the steady (Q_{ss}) and nonsteady (Q_h) discharge
5 simulations. The nonsteady hydrograph (Q_h) lasts longer through time than steady-state discharge (Q_{ss}). If a constant, effective precipitation rate is applied for long enough in the model, the OverlandFlow results will eventually reach this steady discharge value predicted by Eq. (10).

The steady-state hydrology assumption can be problematic when applied to physical systems. Steady-state hydrology is reached when precipitation falls over the entire watershed for long enough duration that water from the furthest upstream point
10 has enough travel time to reach the outlet. This condition will not be met when storms are very short, watersheds have a large drainage area, or both. Under these conditions, predicted steady-state discharge may not be reached in a watershed.

The implementation of the OverlandFlow component in Landlab allows us to investigate the impact of storm characteristics on the resulting hydrograph and how these hydrographs drive erosion processes throughout the watershed. Here, we demonstrate the abilities of this new component and how the component resolves the detail of the storm hydrographs, com-
15 paring them to the traditional landscape evolution hydrology methods. Additionally, in coupling this new component with the Landlab DetachmentLtdErosion component, these model results can illustrate the erosion magnitudes and patterns in response to that hydrograph, and allow us to make inferences about how this type of hydrodynamic model could impact longer term geomorphic evolution of similar watersheds.

7.2 Methods: domain and parameterization

20 To test the new Landlab OverlandFlow component, two synthetic watersheds were generated using the Landlab FlowRouter and StreamPowerEroder components (not described here, see Hobley et al., in review). These basins were evolved to topographic, or geomorphic, steady state, where rock uplift is matched by erosion at all grid locations, and topography is effectively unchanging through time. Two watershed shapes were modeled: a 'square' watershed (Fig. 8a) and a 'long' watershed (Fig. 8b). Each watershed has a drainage area of approximately 36 km^2 at the outlet. The square basin has dimensions of 200 rows by 200
25 columns; the long basin has dimensions of 400 rows by 100 columns. Cells are square and have a resolution of $\Delta x = \Delta y = 30$ m. Each basin has an OPEN_BOUNDARY for the watershed outlet, located at the center node of the southernmost grid edge. The remaining southern nodes, along with the west, east and north grid edges, were set to CLOSED_BOUNDARY status.

To incite flow and incision, three precipitation events were modeled across both watersheds. The base storm, following the example in the real landscape, has a rainfall intensity of 5 mm hr^{-1} falling over 2 hr. To test the impacts of changing intensity
30 and duration on model output, duration was extended compared to the base case (the 'longer duration' storm, Table 3) and intensity was increased relative to the base storm (the 'higher intensity' storm, Table 3). The storm with the longer duration maintained the 5 mm hr^{-1} rainfall intensity, but duration was doubled to 4 hr. In the higher intensity storm, rainfall rate was doubled to 10 mm hr^{-1}, while the base duration of 2 hr was kept.

Discharge was recorded at all points throughout the watershed for each model run. To capture the entire overland flow event, all simulations were run for at least 24 modeled hours. A single 'base' storm on the square watershed run for 24 modeled hours took approximately 80 seconds on a 2014 iMac with 4 GHz Intel Core i7 processors.

The OverlandFlow results from the two test basins (Fig. 8) were coupled with the DetachmentLtdErosion component in Landlab to test the impact of nonsteady hydrology on erosional patterns. At each time step, the DetachmentLtdErosion component calculated total incision depth using Eq. (6). Cumulative incision depth at the end of each modeled run was saved for all grid locations. Both test basins were evolved to topographic steady-state, and so the predicted geomorphic 'steady-state' incision rate is equal to the rock uplift rate used in the StreamPowerEroder component. Total incised depth for the hydrologic steady-state runs can be inferred from this steady-state incision rate. To test the erosional impact of nonsteady hydrology, short-term landscape evolution simulations were run on each basin, for the three precipitation events (Table 3). The known steady-state incision rate and depth can be compared to the predicted DetachmentLtdErosion depth when coupled with the OverlandFlow component. In each basin, an annual precipitation rate of 0.5 m yr^{-1} was set, and each simulation was run for 10 model years. Because of differences in intensity and duration, the base storm was run 50 times, while the longer duration and high intensity storms were run 250 times to achieve 5 m total rainfall depth over 10 years. For each model run, total incised depth was saved at all grid locations.

7.3 Results and implications

The hydrograph measured at the outlet of both the square and long basins are compared with the steady-state hydrographs (Fig. 9). The gray box represents the steady-state case, which produces the same discharge in both watersheds, as they have the same drainage area. In the nonsteady method, hydrograph shapes are distinct between the different basins. In both test basin results from the base case storm (Table 3), the hydrographs persist after precipitation and steady-state discharge end. In the case of the square basin, peak discharge exceeds that predicted by the steady-state case (~ 50 m^3/s), a signal not seen in the long basin results. In the long basin, a singular peak discharge is not clear, but all points in the hydrograph are much less than the predicted steady-state. Because flow in the long basin has to travel a greater distance from the upstream portion of the watershed, there is an elongated hydrograph with no clear peak discharge.

In addition to basin shape, the OverlandFlow component is also sensitive to changes in rainfall characteristics in both test basins. In the square basin, extending the duration of the storm (green line, Fig. 9b) results in a higher overall peak discharge when compared to the base storm (light blue line, Fig. 9b), as well as a longer overall hydrograph. The second peak in the longer duration hydrograph is due to the drainage organization in the square basin (Fig. 8a), when flow from other tributaries reaches the outlet after the initial flood peak (see supplemental video). Increasing the rainfall intensity in the square basin (dark blue line, Fig. 9c) increases peak discharge when compared to the base storm case.

In the square basin, each storm has a clear hydrograph signature. These patterns are distinct from the long basin results. In the long basin, all three storm hydrographs have lower discharges than similar storms in the square basin, reflecting the patterns from the basin comparison plot (Fig. 9a). The higher intensity storm run (mauve line, Fig. 9e), has higher discharge values than both the base case and longer duration runs (Fig. 9d), similar to what was seen in the square basin. However, the hydrograph

shapes and discharge values are largely similar in all long basin cases, with longer, lower hydrograph shapes that reflect the longer travel time of water in the basin.

The nonsteady incision results also demonstrate distinct patterns when compared to the value predicted by geomorphic steady-state. Figure 10 shows that the coupled steady-state hydrology and stream power solutions predict higher incision rates

5 than the nonsteady method at all drainage areas. These patterns are clear in both the long watershed with a broad hydrograph, and the square basin with a more peaked hydrograph. The depth of total incision both basins are on the same order of magnitude, and the pattern of increasing incision depth moving downstream is also similar in both basins (Fig. 10a). While the steady-state topography maintains the same land surface elevation, changing the hydrologic regime to nonsteady would lead to more relief in modeled landscapes, as the downstream will initially erode more rapidly than the upstream channels. In other words, the

10 upstream locations will need to steepen more than the downstream locations in order to reach geomorphic steady-state incision rates at all locations in the landscape. Because the upstream locations must steepen more than the downstream locations in order to reach that geomorphic steady-state, this will also lead to increased channel concavity on landscape evolution timescales.

The pattern of increasing downstream erosion is seen in all storm cases (Figs. 10b,c; Table 3). In both basins, total eroded depth is least in the higher intensity storm, increases in the longer duration storm, and is greatest in the base case. The higher

15 intensity storm exhibits a greater peak discharge, but there are fewer overall higher intensity and longer duration storms when compared to the base storm case to maintain the 5 m total rainfall depth over 10 years. Additionally, when calculating total incision using the stream power model, increases in discharge are less significant than the water surface slope due to the exponents m and n. While not explored here, changing the stream power exponents m and n will likely impact the steady and nonsteady fluvial erosion results in this model.

20 Overall, these results suggest that when compared to the OverlandFlow component, hydrologic steady-state predictions can over- or underestimate the peak of a hydrograph depending on basin orientation or shape (Fig. 9a). Additionally, the hydrodynamic algorithm from de Almeida et al. (2012) is sensitive to rainfall inputs, both with changes in duration and intensity (Figs. 9b-e). This component can be applied across a range of time scales, used for predictions of overland flow for a single storm or multiple storms, and used efficiently with other process components in Landlab, as demonstrated by coupling

25 to the DetachmentLtdErosion component.

The patterns of erosion support earlier findings by Sólyom and Tucker (2004), which suggested that landscapes dominated by nonsteady runoff patterns can be characterized by greater overall relief. Their results were generated using an incision rate controlled by the peak discharge. While the runs using the Landlab model were over shorter timescales, these results were integrated over the entirety of the hydrograph, not just the peak discharge. These results suggest that on longer timescales,

30 watershed morphology would vary depending on the method used to calculate overland flow. Additionally, as the watershed morphology evolves in response to these spatial variations in incision rate, the hydrograph shape may change, impacting overall incision patterns and rates. The difference in patterns between steady and nonsteady hydrology implies that the retention of water within the watershed during a runoff event, driven by nonsteady hydrology, can drive have morphological significance over longer-term landscape evolution.

8 Future applications

Post-wildfire hydrologic changes have been linked to large post-fire erosion events. The Landlab OverlandFlow model could be used to explore the processes driving post-fire flooding and erosion. For example, this model has been applied to explore trends in discharge and sediment yield in the Spring Creek watershed (Adams et al., 2016). To fully capture post-fire dynamics, this model can be coupled with other components, such as infiltration or transport-limited sediment transport. In landscapes with extensive field data, both pre- and post-fire, the model can be used to understand potential post-fire responses to large storm events.

Another application under exploration is a model created by coupling the Overland Flow component with Landlab's ecohydrology components (Nudurupati et al., 2015). In this model type, OverlandFlow can be used to drive water discharge and update water depths across the surface. Surface water depths can be used to simulate infiltration in the SoilInfiltrationGreenAmpt component. The SoilMoisture component computes the water balance and root-zone soil moisture values. Soil moisture can drive changes in the Vegetation component, which simulates above-ground live and dead biomass. This coupled model can be used to study how differences in steady- and nonsteady hydrology may drive vegetation evolution on annual to decadal timescales.

9 Conclusions

The OverlandFlow component successfully integrates a two-dimensional hydrodynamic algorithm into the Landlab modeling framework. Being part of the Landlab modeling framework comes with many advantages. The OverlandFlow component can make use of DEM input and output utilities and be coupled with other Landlab components, illustrated here as a distributed rainfall-runoff model. It also can be coupled to the stream power DetachmentLtdErosion component to explore decadal impacts of a hydrograph on erosion. The OverlandFlow component is also flexible enough to allow for future applications in both geomorphology and hydrology.

This manuscript illustrates the theory in the OverlandFlow component in Landlab and how to use and couple the component to other Landlab components. In the synthetic landscapes explored here, the hydrograph results from the OverlandFlow component demonstrate a sensitivity to both basin shape, precipitation duration and intensity. The erosion results predicted by using steady-state and nonsteady hydrology are distinct in both the patterns and magnitudes of eroded depth and incision rates. Incision driven by nonsteady hydrology showed increasing incision rates moving downstream in the modeled watersheds. These results suggest that nonsteady hydrology could have important implications for predicting watershed relief and hypsometry in landscapes with different rainfall regimes, and that choice of hydrology method can have implications for both short- and long-term landscape evolution modeling results.

30 10 Code availability

The Landlab OverlandFlow and DetachmentLtdErosion components are part of Landlab version 1.0.0. Source code for the Landlab project is housed on GitHub: http://github.com/landlab/landlab. Documentation, installation instructions and software dependencies for the entire Landlab project can be found at: http://landlab.github.io/. Driver scripts for the applications can be found at: https://github.com/landlab/pub_adams_etal_gmd (Adams, GitHub Repository). The Landlab project is tested on recent-generation Mac, Linux and Windows platforms using Python versions 2.7, 3.4, and 3.5. The Landlab modeling frame-
5 work is distributed under a MIT open-source license.

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. This research was supported by the National Science Foundation grants ACI-1147519 and ACI-1450338 (PI: Gasparini), ACI-1148305 and ACI-1450412 (PI: Istanbulluoglu), ACI-1147454 (PI: Tucker) and ACI-1450409 (PI: Tucker, Co-PI: Hobley), as well as the Tulane University Department of Earth and Environmental Sciences Vokes Fellowship.

10 References

Adams, J. M.: GitHub Repository: OverlandFlow example drivers and documentation, doi:10.5281/zenodo.162058.

Adams, J. M., Nudurupati, S. S., Gasparini, N. M., Hobley, D. E., Hutton, E. W. H., Tucker, G. E., and Istanbulluoglu, E.: Landlab: Sustainable software development in practice., Technical Report 1097629, figshare, doi:dx.doi.org/10.6084/m9.figshare.1097629, 2014.

Adams, J. M., Gasparini, N. M., Hobley, D. E., Tucker, G. E., Hutton, E. W. H., Nudurupati, S. S., and Istanbulluoglu, E.: Flooding and erosion after the Buffalo Creek fire: a modeling approach using Landlab, Presented at the Geological Society of America Annual Meeting,
5   Denver, CO, USA, 2016.

Aksoy, H. and Kavvas, M.: A review of hillslope and watershed scale erosion and sediment transport models, Catena, 64, 247–271, doi:10.1016/j.catena.2005.08.008, 2005.

Anders, A. M., Roe, G. H., Montgomery, D. R., and Hallet, B.: Influence of precipitation phase on the form of mountain ranges, Geology, 36, 479–482, doi:10.1130/G24821A.1, 2008.

10 Attal, M., Cowie, P., Whittaker, A., Hobley, D., Tucker, G., and Roberts, G. P.: Testing fluvial erosion models using the transient response of bedrock rivers to tectonic forcing in the Apennines, Italy, Journal of Geophysical Research-Earth Surface, 116, doi:10.1029/2010JF001875, 2011.

Bates, P. D. and De Roo, A. P. J.: A simple raster-based model for flood inundation simulation, Journal of Hydrology, 236, 54–77, doi:10.1016/S0022-1694(00)00278-X, 2000.

15 Bates, P. D., Horritt, M. S., and Fewtrell, T. J.: A simple inertial formulation of the shallow water equations for efficient two-dimensional flood inundation modelling, Journal of Hydrology, 387, doi:10.1016/j.jhydrol.2010.03.027, 2010.

Beeson, P. C., Martens, S. N., and Breshears, D. D.: Simulating overland flow following wildfire: mapping vulnerability to landscape disturbance, Hydrological Processes, 15, 2917–2930, doi:10.1002/hyp.382, 2001.

Beven, K. and Kirkby, M. J.: A physically based, variable contributing area model of basin hydrology, Hydrological Sciences Journal, 24,
20 43–69, doi:10.1080/02626667909491834, 1979.

Beven, K. J.: Rainfall-runoff modelling: the primer, John Wiley & Sons, 2011.

Braun, J. and Sambridge, M.: Modelling landscape evolution on geological time scales: a new method based on irregular spatial discretization, Basin Research, 9, 27–52, doi:10.1046/j.1365-2117.1997.00030.x, 1997.

Chen, A., Darbon, J., and Morel, J. M.: Landscape evolution models: A review of their fundamental equations, Geomorphology, 219, 68–86,
25 doi:10.1016/j.geomorph.2014.04.037, 2014.

Colberg, J. and Anders, A.: Numerical modeling of spatially-variable precipitation and passive margin escarpment evolution, Geomorphology, 207, 203–212, doi:10.1016/j.geomorph.2013.11.006, 2014.

Coulthard, T. J.: Landscape evolution models: a software review, Hydrol. Process, 15, 165–173, doi:10.1002/hyp.426, 2001.

Coulthard, T. J., Macklin, M. G., and Kirkby, M. J.: A cellular model of Holocene upland river basin and alluvial fan evolution, Earth Surface
30 Processes and Landforms, 27, 269–288, doi:10.1002/esp.318, 2002.

Coulthard, T. J., Neal, J. C., Bates, P. D., Ramirez, J., de Almeida, G. A. M., and Hancock, G. R.: Integrating the LISFLOOD-FP 2D hydrodynamic model with the CAESAR model: implications for modelling landscape evolution, Earth Surface Processes and Landforms, 38, 1897–1906, doi:10.1002/esp.3478, 2013.

de Almeida, G. A. M. and Bates, P.: Applicability of the local inertial approximation of the shallow water equations to flood modeling, Water
35 Resources Research, 49, 4833–4844, doi:10.1002/wrcr.20366, 2013.

de Almeida, G. A. M., Bates, P., Freer, J. E., and Souvignet, M.: Improving the stability of a simple formulation of the shallow water equations for 2-D flood modeling, Water Resources Research, 48, doi:10.1029/2011wr011570, 2012.

Devi, G., Gansari, B., and Dwarakish, G.: A review on hydrological models, Aquatic Procedia, 4, 1001–1007, doi:10.1016/j.aqpro.2015.02.126, 2015.

DiBiase, R. A. and Whipple, K. X.: The influence of erosion thresholds and runoff variability on the relationships among topography, climate, and erosion rate, Journal of Geophysical Research-Earth Surface, 116, doi:10.1029/2011JF002095, 2011.

5 Donigan, A. S., Imhoff, J. C., Bicknell, B. R., and Kittle, J. L.: Application Guide for Hydrological Simulation Program FORTRAN(HSPF): U.S. Environmental Protection Agency, Environmental Research Laboratory, Athens, GA., EPA-600/3-84-065, 1984.

Downer, C. W. and Ogden, F. L.: GSSHA: Model to simulate diverse stream flow producing processes, Journal of Hydrologic Engineering, 9, 161–174, doi:10.1061/(ASCE)1084-0699(2004)9:3(161), 2004.

Dutta, D., Herath, S., and Musiake, K.: Flood inundation simulation in a river basin using a physically based distributed hydrologic model, Hydrological Processes, 14, 497–519, doi:10.1002/(SICI)1099-1085(20000228)14:3<497::AID-HYP951>3.0.CO;2-U, 2000.

Esteves, M., Faucher, X., Galle, S., and Vauclin, M.: Overland flow and infiltration modelling for small plots during unsteady rain: numerical results versus observed values, Journal of Hydrology, 228, 265–282, doi:10.1002/(SICI)1099-1085(20000228)14:3<497::AID-HYP951>3.0.CO;2-U, 2000.

Francipane, A., Ivanov, V. Y., Noto, L. V., Istanbulluoglu, E., Arnone, E., and Bras, R. L.: tRibs-Erosion: A parsimonious physically-based 15 model for studying catchment hydro-geomorphic response., Catena, 92, 216–231, doi:10.1016/j.catena.2011.10.005, 2012.

Grant, G. E.: Critical flow constrains flow hydraulics in mobile-bed streams: A new hypothesis, Water Resources Research, 33, 349–358, doi:10.1029/96WR03134, 1997.

Han, J., Gasparini, N. M., and Johnson, J. P.: Measuring the imprint of orographic rainfall gradients on the morphology of steady-state numerical fluvial landscapes, Earth Surface Processes and Landforms, 40, 1334–1350, doi:10.1002/esp.3723, 2015.

20 Hancock, G. R., Lowry, J. B. C., and Coulthard, T. J.: Catchment reconstruction - erosional stability at millennial time scales using landscape evolution models, Geomorphology, 231, 15–27, doi:10.1016/j.geomorph.2014.10.034, <GotoISI>://WOS:000348957100002, 2015.

Hobley, D. E. J., Adams, J. M., Nudurupati, S. S., Hutton, E. W. H., Gasparini, N. M., Istanbulluoglu, E., and Tucker, G. E.: Creative computing with Landlab: an open-source toolkit for building, coupling, and exploring two-dimensional numerical models of Earth-surface dynamics, Earth Surface Dynamics Discussion, doi:10.5194/esurf-2016-45.

25 Horritt, M. and Bates, P.: Evaluation of 1D and 2D numerical models for predicting river flood inundation, Journal of Hydrology, 268, 87–99, doi:10.1002/hyp.188, 2002.

Howard, A. D.: A detachment-limited model of drainage-basin evolution, Water Resources Research, 30, 2261–2285, doi:10.1029/94wr00757, 1994.

Huang, X. and Niemann, J.: Simulating the impacts of small convective storms and channel transmission losses on gully evolution, Reviews 30 in Engineering Geology, 22, 131 – 145, doi:10.1130/2014.4122(13), 2014.

Hunter, N. M., Horritt, M. S., Bates, P. D., Wilson, M. D., and Werner, M. G. F.: An adaptive time step solution for raster-based storage cell modelling of floodplain inundation, Advances in Water Resources, 28, 975–991, doi:10.1016/j.advwatres.2005.03.007, 2005.

Hunter, N. M., Bates, P. D., Horritt, M. S., and Wilson, M. D.: Simple spatially-distributed models for predicting flood inundation: a review, Geomorphology, 90, 208–225, doi:10.1016/j.geomorph.2006.10.021, 2007.

35 Ivanov, V. Y., Vivoni, E. R., Bras, R. L., and Entekhabi, D.: Catchment hydrologic response with a fully distributed triangulated irregular network model, Water Resources Research, 40, doi:10.1029/2004wr003218, 2004.

Kazezyılmaz-Alhan, C. M. and Medina Jr, M. A.: Kinematic and diffusion waves: analytical and numerical solutions to overland and channel flow, Journal of Hydraulic Engineering, 133, 217–228, doi:10.1061/(ASCE)0733-9429(2007)133:2(217), 2007.

Kim, J., Ivanov, V. Y., and Katopodes, N. D.: Modeling erosion and sedimentation coupled with hydrological and overland flow processes at the watershed scale, Water Resources Research, 49, 5134–5154, doi:10.1002/wrcr.20373, 2013.

Kollet, S. J. and Maxwell, R. M.: Integrated surface–groundwater flow modeling: A free-surface overland flow boundary condition in a parallel groundwater flow model, Advances in Water Resources, 29, 945–958, doi:10.1016/j.advwatres.2005.08.006, 2006.

5 Lague, D., Hovius, N., and Davy, P.: Discharge, discharge variability, and the bedrock channel profile, Journal of Geophysical Research-Earth Surface, 110, doi:10.1029/2004jf000259, 2005.

Molnar, P., Anderson, R. S., Kier, G., and Rose, J.: Relationships among probability distributions of stream discharges in floods, climate, bed load transport, and river incision, Journal of Geophysical Research-Earth Surface, 111, doi:10.1029/2005jf000310, 2006.

Moradkhani, H. and Sorooshian, S.: General review of rainfall-runoff modeling: model calibration, data assimilation, and uncertainty anal-
10 ysis, in: Hydrological modelling and the water cycle: coupling the atmospheric and hydrological models, edited by Sorooshian, S., Hsu, K.-L., Coppola, E., Tomassetti, B., Verdecchia, M., and Visconti, G., pp. 1–24, Springer Berlin Heidelberg, Berlin, Heidelberg, doi:10.1007/978-3-540-77843-1_1, 2009.

Nudurupati, S. S., Istanbulluoglu, E., Adams, J. M., Hobley, D. E., Gasparini, N. M., Tucker, G. E., and Hutton, E. W. H.: Elevation control on vegetation organization in a semiarid ecosystem in Central New Mexico, Presented at the American Geophysical Union Fall Meeting,
15 San Francisco, CA, USA, 2015.

Ogden, F., Julien, P., Singh, V., and Frevert, D.: CASC2D: A two-dimensional, physically-based, Hortonian hydrologic model., in: Mathematical models of small watershed hydrology and applications, pp. 69–112, Water Resources Publications, 2002.

Peckham, S. D.: The CSDMS standard names: cross-domain naming conventions for describing process models, data sets and their associated variables, in: Proceedings of the 7th International Congress on Environmental Modeling and Software, edited by Daniel P. Ames, Nigel
20 W. T. Quinn, A. E. R., International International Environmental Modelling and Software Society (IEMSs), San Diego, California, USA, 2014.

Pelletier, J. D.: Persistent drainage migration in a numerical landscape evolution model, Geophysical Research Letters, 31, doi:10.1029/2004gl020802, 2004.

Rengers, F., McGuire, L., Kean, J. W., Staley, D. M., and Hobley, D.: Model simulations of flood and debris flow timing in steep catchments
25 after wildfire, Water Resources Research, doi:10.1002/2015WR018176, 2016.

Scharffenberg, W. A. and Fleming, M. J.: Hydrologic Modeling System HEC-HMS: User's Manual, US Army Corps of Engineers, Hydrologic Engineering Center, 2006.

Sólyom, P. B. and Tucker, G. E.: Effect of limited storm duration on landscape evolution, drainage basin geometry, and hydrograph shapes, Journal of Geophysical Research-Earth Surface, 109, doi:10.1029/2003jf000032, 2004.

30 Tarboton, D. G.: A new method for the determination of flow directions and upslope areas in grid digital elevation models, Water Resources Research, 33, 309–319, doi:10.1029/96WR03137, 1997.

Tucker, G.: Lidar dataset, aquisition and processing: National Center for Airborne Laser Mapping (NCALM), doi:10.5069/G9TM782F, 2010.

Tucker, G. E. and Bras, R. L.: A stochastic approach to modeling the role of rainfall variability in drainage basin evolution, Water Resources
35 Research, 36, 1953–1964, doi:10.1029/2000WR900065, 2000.

Tucker, G. E. and Hancock, G. R.: Modelling landscape evolution, Earth Surface Processes and Landforms, 35, 28–50, doi:10.1002/esp.1952, 2010.

Tucker, G. E. and Slingerland, R. L.: Erosional dynamics, flexural isostasy, and long-lived escarpments - a numerical modeling study, Journal of Geophysical Research-Solid Earth, 99, 12 229–12 243, doi:10.1029/94jb00320, 1994.

Tucker, G. E., Lancaster, S. T., Gasparini, N. M., Bras, R. L., and Rybarczyk, S. M.: An object-oriented framework for distributed hydrologic and geomorphic modeling using triangulated irregular networks, Computers & Geosciences, 27, 959–973, doi:10.1016/S0098-3004(00)00134-5, 2001.

Tucker, G. E., Hobley, D. E., Hutton, E. W. H., Gasparini, N. M., Istanbulluoglu, E., Adams, J. M., and Nudurupati, S. S.: CellLab-CTS 2015: a Python library for continuous-time stochastic cellular automaton modeling using Landlab., Geoscientific Model Development, 8, 9507 – 9552, doi:10.5194/gmd-9-823-2016, 2016.

Whipple, K. X. and Tucker, G. E.: Dynamics of the stream-power river incision model: Implications for height limits of mountain ranges, landscape response timescales, and research needs, Journal of Geophysical Research-Solid Earth, 104, 17 661–17 674, doi:10.1029/1999jb900120, 1999.

Whipple, K. X. and Tucker, G. E.: Implications of sediment-flux-dependent river incision models for landscape evolution, Journal of Geophysical Research-Solid Earth, 107, doi:10.1029/2000jb000044, 2002.

Willgoose, G.: A statistic for testing the elevation characteristics of landscape simulation-models, Journal of Geophysical Research-Solid Earth, 99, 13 987–13 996, doi:10.1029/94jb00123, 1994.

Willgoose, G.: Mathematical modeling of whole landscape evolution, Annual Review of Earth and Planetary Sciences, 33, 443–459, doi:10.1146/annurev.earth.33.092203.122610, 2005.

Willgoose, G., Bras, R. L., and Rodriguez-Iturbe, I.: A coupled channel network growth and hillslope evolution model 1. Theory, Water Resources Research, 27, 1671–1684, doi:10.1029/91wr00935, 1991.

Woolhiser, D. A., Smith, R., and Goodrich, D. C.: KINEROS: a kinematic runoff and erosion model: documentation and user manual, US Department of Agriculture, Agricultural Research Service, 1990.

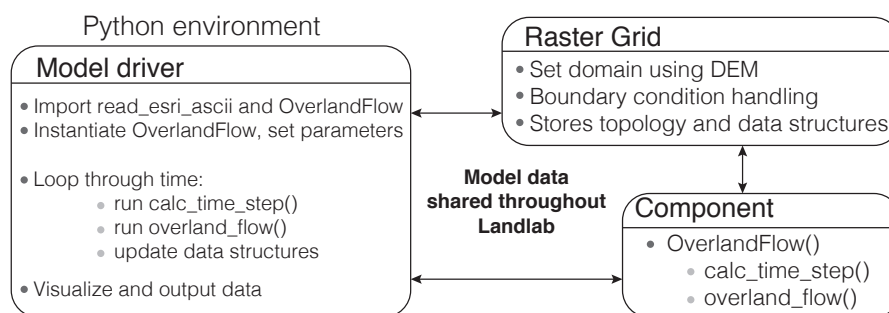



Figure 1. Sample workflow for the Landlab OverlandFlow component. Users create or use a pre-developed model driver, where the grid, components and model utilities are imported and instantiated. The time loop is set in the driver, and at each time step the component methods are called and the data structures are updated.

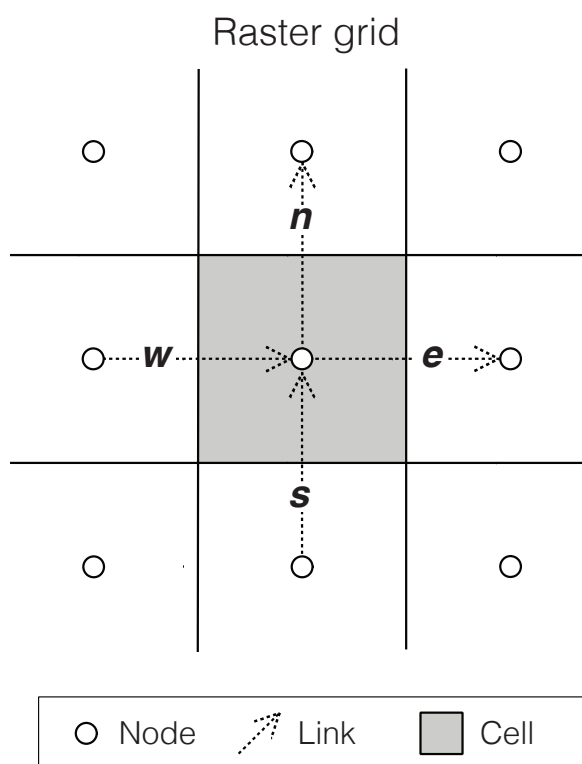



Figure 2. Example of the Landlab structured grid type with key topological elements shown. In the Landlab OverlandFlow component, RasterModelGrid class stores data at both nodes and links. Links denoted as west (w) and south (s) are called "inlinks", while north (n) and east (e) are "outlinks". Direction is only for topological reference; flux directionality is tied to gradients on the grid.

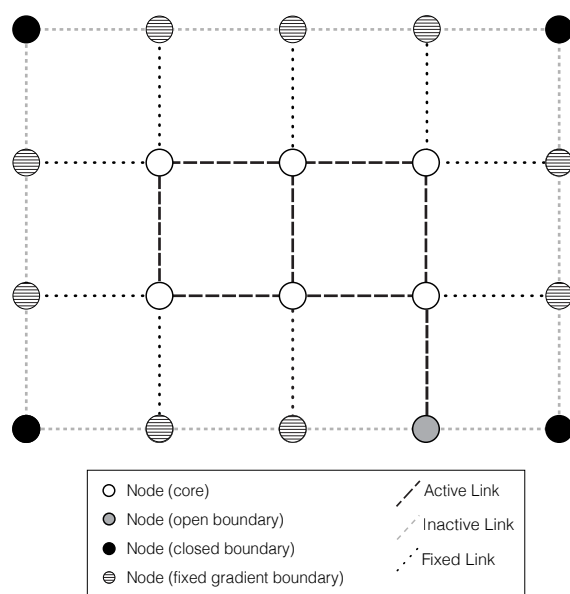



Figure 3. Simple example of Landlab RasterModelGrid, demonstrating both node and link boundary conditions. The OverlandFlow class calculates fluxes at active links, and can update the surrounding fixed links according to these fluxes. No fluxes are calculated at inactive links. Water depth is updated at core and open boundary nodes. No calculations are performed on closed or fixed gradient boundaries. Note that RasterModelGrid cell elements are not illustrated here.

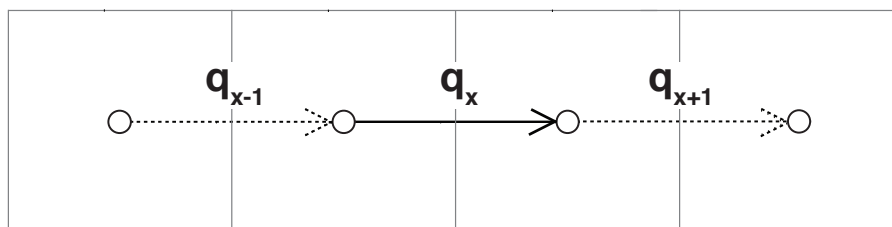



Figure 4. In the de Almeida et al. (2012) equation, flux information from neighboring links is used to calculate surface water discharge. In this sample one-dimensional grid, discharge is calculated in the horizontal (subscript x) direction on links. Here, discharge is calculated at location q_x using the left neighbor (q_{x-1}) and right neighbor (q_{x+1}), following Eq. (3).

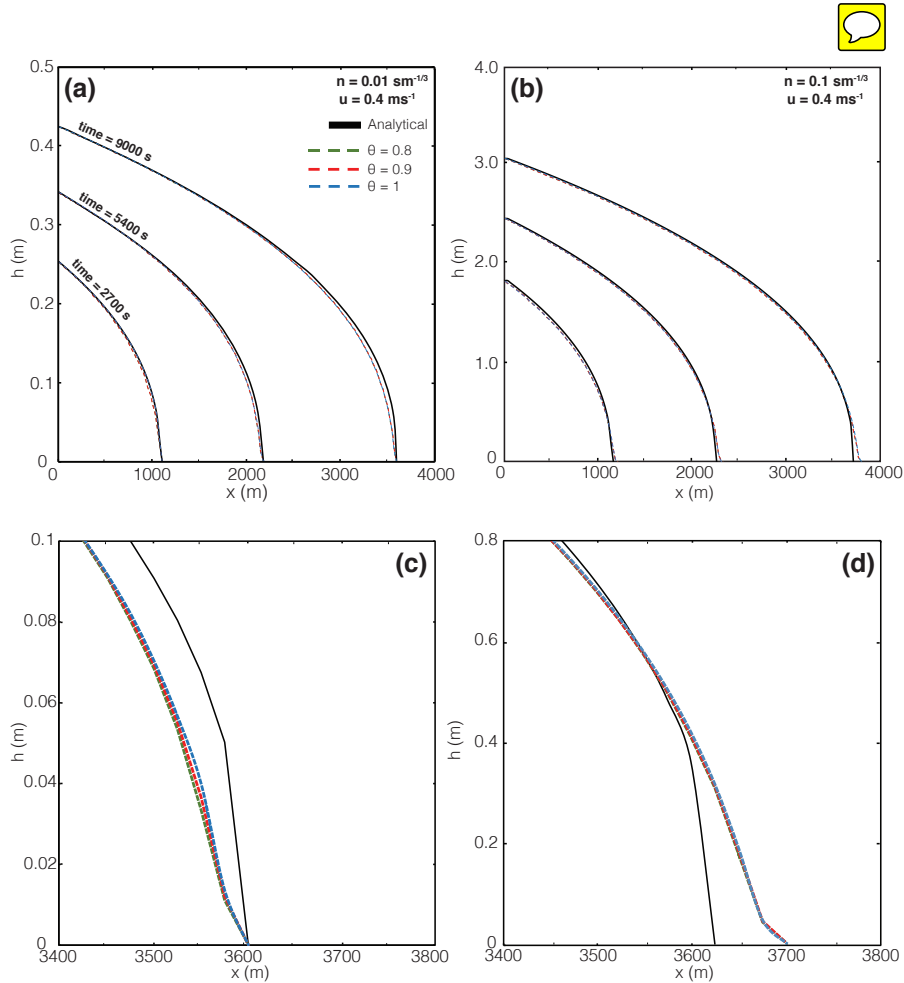



Figure 5. Results from test of analytical solution using the Landlab OverlandFlow component, illustrated in the same manner as Fig. (2) from de Almeida et al. (2012). Water depth was plotted against distance for two combinations of velocity and friction coefficient values. Both panels **(a)** and **(b)** show water depths for $t = 2700$, 5400, and 9000 s. Panels **(c)** and **(d)** are zoomed-in images of the wave front at time 9000 s.

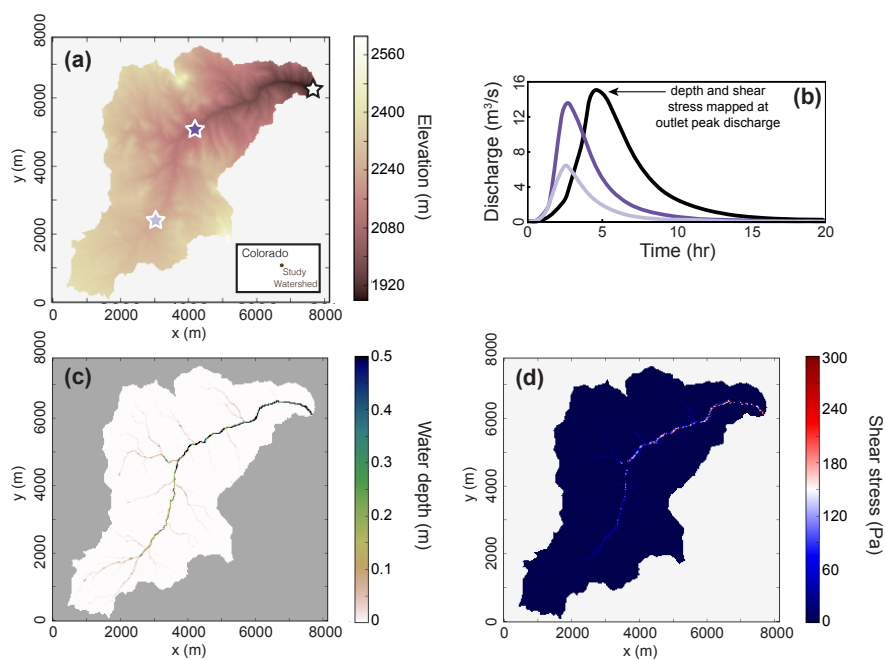



Figure 6. Results from the real landscape application. Panel **(a)** shows the topography of the Spring Creek watershed, and the inset notes the location of this watershed in central Colorado, USA. Panel **(b)** illustrates the hydrographs from three points within the watershed. The location for each hydrograph sampling site is shown in panel **(a)**, with the lightest color at the upstream, darkening in color towards the outlet. Panel **(c)** shows the water depth, and panel **(d)** the shear stress, both plotted at the peak of this hydrograph, as noted by the arrow in panel **(b)**. Note that the discontinuities in the shear stress figure are a result of the uneven bed topography, and variations in the surface water slope linked to that topography.

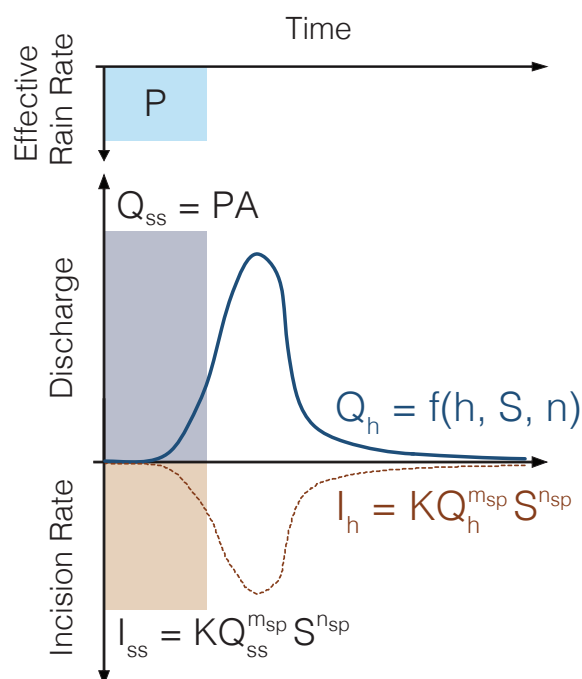



Figure 7. Cartoon illustrating the differences between steady-state and nonsteady hydrology and incision at a single point within a watershed. In this schematic, the effective precipitation rate P is the same for both steady and nonsteady cases. During the precipitation event, steady discharge Q_{ss} and incision rate I_{ss} are constant, whereas in the nonsteady case, a wave front begins to propagate and incise, producing time varying discharge Q_h and incision rate I_h. At the end of the precipitation event, Q_{ss} and I_{ss} also end, while nonsteady values Q_h and I_h continue until all water has completely exited the system at the outlet.

(a) Square basin topography

(b) Long basin topography

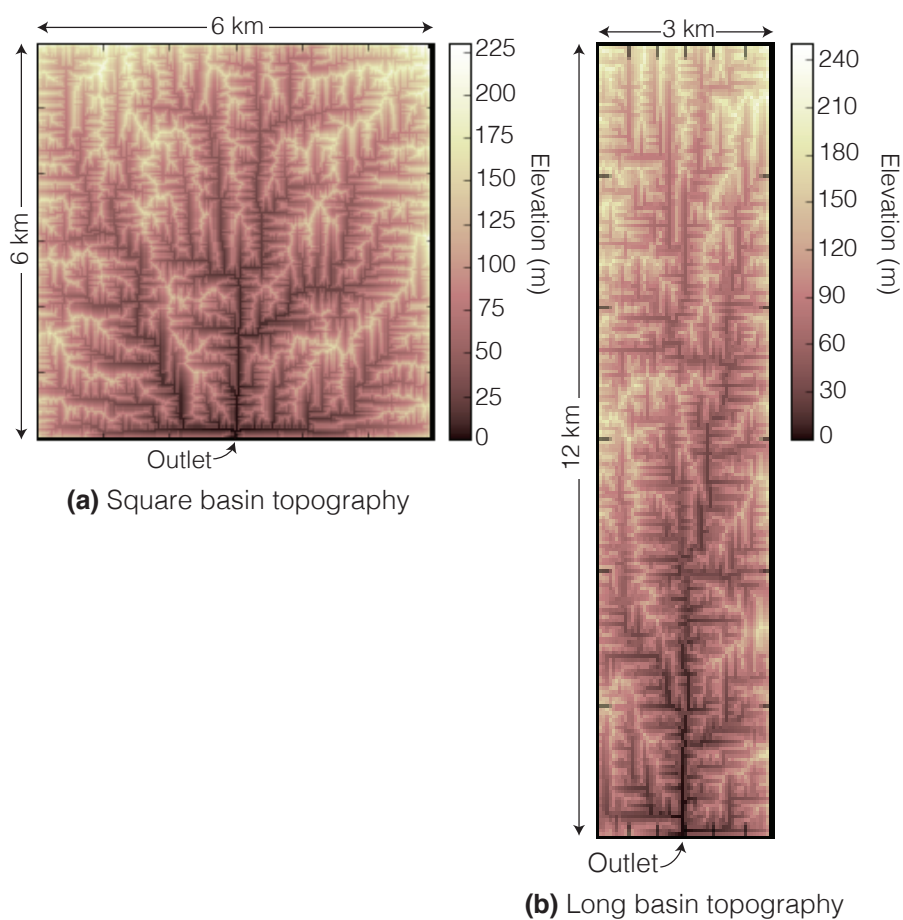



Figure 8. Two test basins evolved using the Landlab FlowRouter and StreamPowerEroder components, generating a network using D4 flow routing and erosion methods. Each grid was evolved from an initial random topography to steady-state, where uplift rate is matched by incision rate. Both basins have the same drainage area (36 km^2) at the watershed outlet, but different dimensions: panel **(a)** 200 rows x 200 columns, and panel **(b)** 400 rows x 100 columns. Note the perpendicular junctions are due to the D4 flow routing scheme.

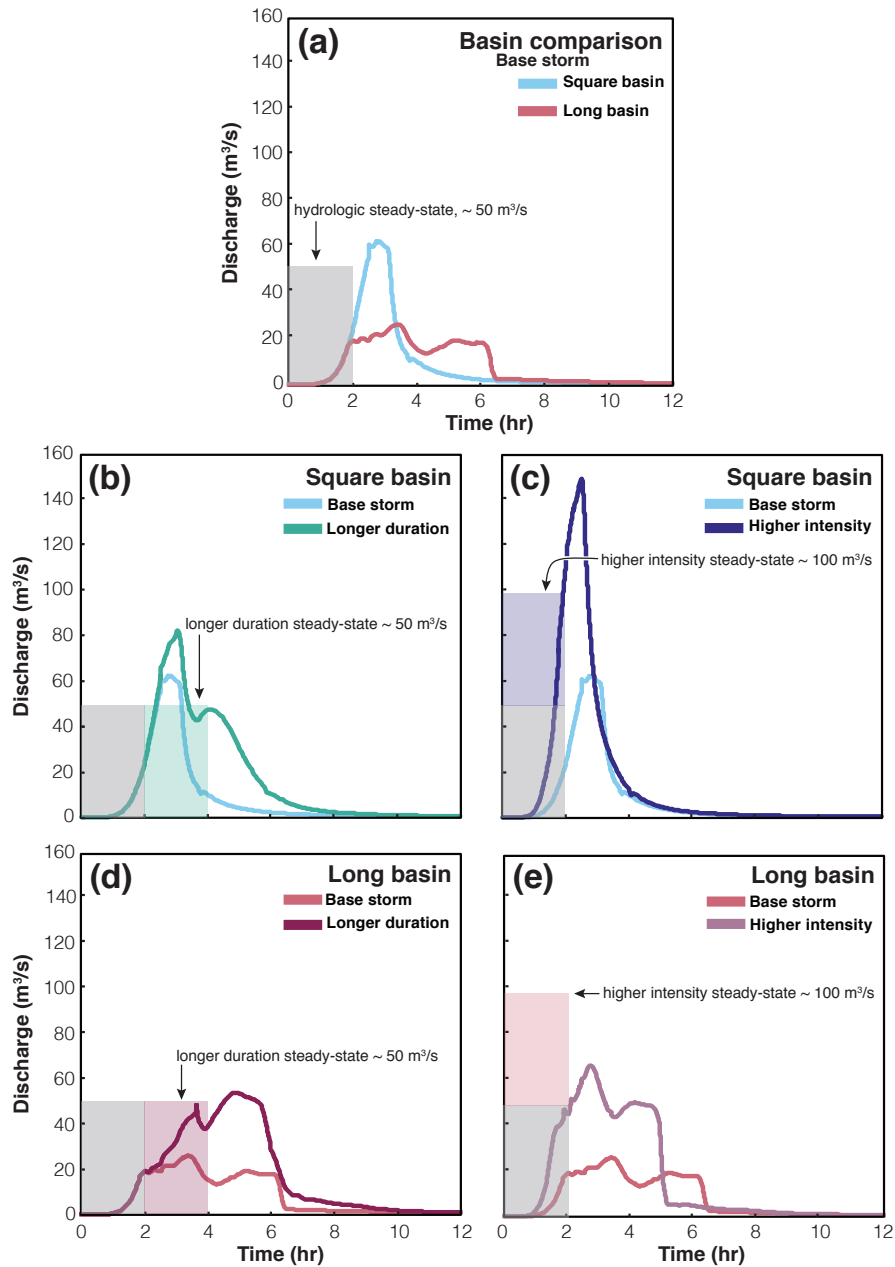



Figure 9. OverlandFlow output for all storms described in Table (**3**). Hydrographs are taken from the active link upstream of the outlet node. Steady-state discharge is shown for each event, with the gray box representing the base storm in all cases. Panel (**a**) shows the base storm for both the square basin and the long basin; panel (**b**) compares outlet hydrographs from the base and longer duration storms in the square basin; panel (**c**) compares outlet hydrographs from the base and higher intensity storms in the square basin; panel (**d**) compares outlet hydrographs from the base and longer duration storms in the long basin; panel (**e**) compares outlet hydrographs from the base and higher intensity storms in the long basin.

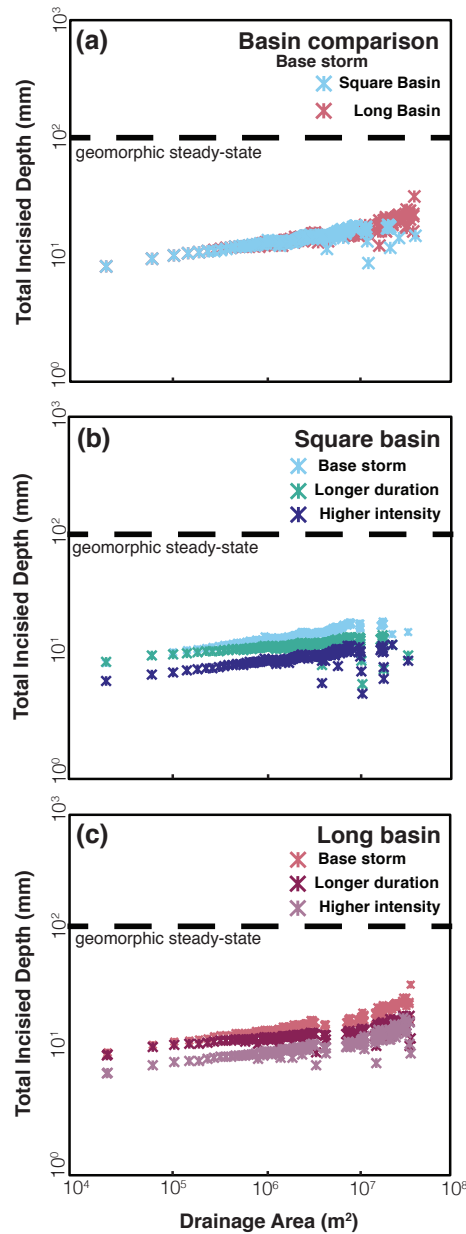



Figure 10. DetachmentLtdErosion output for all storms described in Table (3). Incision depth was taken after 10 years of modeled storms from the OverlandFlow component for all grid locations. The average incision depth was plotted at each drainage area: panel **(a)** shows incision depth versus drainage area for both the square and long basin after 10 years of the base storm; panel **(b)** shows total incision results from the square basin for all three precipitation events after 10 years; and panel **(c)** shows total incision results from the long basin for all three precipitation events after 10 years.

Table 1. List of variables used in the OverlandFlow and DetachmentLtdErosion. For each variable the name, grid element and units are given.

Variable	Name	Grid element	Units
q	water discharge	link	$m^2 s^{-1}$
h_f	local maximum water depth	link	m
S_w	water surface slope	link	–
h	water depth	node	m
Q_h	water discharge from hydrograph method	node	$m^3 s^{-1}$
I	incision rate	node	ms^{-1}
$S_{w_{max}}$	local maximum water surface slope	node	–

Table 2. List of parameters used in the OverlandFlow and DetachmentLtdErosion. For each variable the name and units are given.

Parameter	Name	Default value	Units
Δt	time step	adaptive	s
α	stability coefficient	0.7	–
g	gravity	9.81	ms^{-2}
θ	weighting parameter	0.8	–
n	Manning's n, surface roughness coefficient	0.3	$sm^{-1/3}$
K	erodibility coefficient	$1.26 * 10^{-7}$	$m^{1-2m_{sp}}s^{-1}$
m_{sp}	stream power coefficient	0.5	–
n_{sp}	stream power coefficient	1.0	–
β	entrainment threshold	0.0	ms^{-1}
ρ	fluid density	1000.0	$kg\ m^{-3}$

Table 3. Precipitation parameters for the three storm cases routed across the test basins.

Storm ID	Intensity	Duration
Base Storm	$5.0\ mm\ hr^{-1}$	$2\ hr$
Longer Duration	$5.0\ mm\ hr^{-1}$	$4\ hr$
Higher Intensity	$10\ mm\ hr^{-1}$	$2\ hr$

Algorithm 1 Sample Landlab overland flow and erosion model

1: **from** landlab.components **import** OverlandFlow, DetachmentLtdErosion, SinkFiller   *#Import Landlab components and utilities*

2: **from** landlab.io **import** read_esri_ascii

3: (grid, elevations) = read_esri_ascii(asc_file='watershed_DEM.asc', name='topographic__elevation') *#Read in DEM and create grid*

4: grid.set_watershed_boundary_condition(elevations, nodata_value = -9999.0)   *#Set boundary conditions*

5: effective_rain_rate_ms = 5.0 * (2.78 * 10^{-7})   *#Convert rainfall from mm hr^{-1} to m s^{-1}*

6: dle = DetachmentLtdErosion(grid)   *#Instantiate components and set parameters*

7: of = OverlandFlow(grid, steep_slopes=TRUE, rainfall_intensity = effective_rain_rate_ms)

8: sf = SinkFiller(grid, routing='D4')

9: sf.fill_pits()   *#Pre-process DEM and fill pits in D4 flow-routing scheme*

10: elapsed_time = 0.0   *#Start time in seconds*

11: **while** elapsed_time < 36000.0 :   *#Run for 10 modeled hours*

12:     Δt = calculate_time_step()   *#Calculate stable time step*

13:     of.overland_flow(dt = Δt)   *#Generate overland flow*

 # Below, populate fields with water discharge and water surface slope to be shared across components

14:     grid['node']['surface_water__discharge'] = of.discharge_mapper(of.q, convert_to_volume = True)

15:     grid['node']['water_surface__slope'] = (of.water_surface_slope[grid.links_at_node] * grid.active_link_dirs_at_node).max(axis=1)

16:     dle.erode(dt = Δt, discharge_cms = 'surface_water__discharge', slope = 'water_surface__slope')   *#Erode the landscape*

17:     elapsed_time += Δt *#Updated elapsed time*