Dynamic Landscape Evolution

Brendan Alexander Harmon[a,b,*], Helena Mitasova[a,c], Vaclav Petras[a,c], Anna Petrasova[a,c]

[a] *Center for Geospatial Analytics, North Carolina State University, Raleigh, North Carolina, United States of America*
[b] *Department of Landscape Architecture, North Carolina State University, Raleigh, North Carolina, United States of America*
[c] *Department of Marine, Earth, and Atmospheric Sciences, North Carolina State University, Raleigh, North Carolina, United States of America*

Abstract

This is a fine-scale, short term, process-based landscape evolution model using simulated erosion and deposition to generate a timeseries of digital elevation models. This model uses a path sampling method to solve water and sediment flow continuity equations and model mass flows over complex topographies based on topographic, land cover, soil, and rainfall parameters. This either steady state or dynamic model can simulate landscape evolution for a range of hydrologic soil erosion regimes.

Keywords: landscape evolution, dynamic model

1. Introduction

This process-based, spatial distributed, dynamic model uses a path sampling method to solve the water and sediment flow equations [2] and model mass flows over complex topographies based on topographic, land cover, soil, and rainfall parameters. The modeled flow of sediment – a function of the flow of water, soil detachment, and transport parameters – is then used to estimate the net erosion and deposition rates and the associated short-term evolution of the topography.

1.1. Shallow water flow

We simulated shallow overland water flow controlled by spatially variable topography, soil, landcover, and rainfall parameters using the SIMWE model to solve the continuity and momentum equations for steady state water flow with a path sampling method. Shallow water flow can be approximated by the bivariate form of the St Venant equation:

$$\frac{\partial h(\mathbf{r},t)}{\partial t} = i_e(\mathbf{r},t) - \nabla \cdot \mathbf{q}(\mathbf{r},t) \tag{1}$$

where:

$\mathbf{r}(x,y)$ is the position [m]
t is the time [s]
$h(\mathbf{r},t)$ is the depth of overland flow [m]
$i_e(\mathbf{r},t)$ is the rainfall excess [m/s]
(rainfall − infiltration − vegetation intercept)
$\mathbf{q}(\mathbf{r},t)$ is the water flow per unit width [m^2/s].

By integrating a diffusion term $\propto \nabla^2[h^{5/3}(\mathbf{r})]$ into the solution of the continuity and momentum equations for steady state water flow diffusive wave effects can be approximated so that water can flow through depressions.

$$-\frac{\varepsilon(\mathbf{r})}{2}\nabla^2[h^{5/3}(\mathbf{r})] + \nabla \cdot [h(\mathbf{r})\mathbf{v}(\mathbf{r})] = i_e(\mathbf{r}) \tag{2}$$

where:

$\varepsilon(\mathbf{r})$ is a spatially variable diffusion coefficient.

This equation is solved using a Green's function Monte Carlo path sampling method [2].

1.2. Sediment flow

Steady state sediment flow equation with diffusion...

$$\ldots \tag{3}$$

1.3. Landscape evolution

Detachment limited landscape evolution

$$\Delta z(x,y,t) = \Delta t \cdot q_s(x,y,t) \cdot \varrho(r)^{-1} \tag{4}$$

where:

Δz = change in elevation (m)
q_s = sediment flux ($kg \cdot m^{-1} s^{-1}$)

*Corresponding author
Email addresses: brendan.harmon@gmail.com (Brendan Alexander Harmon), hmitaso@ncsu.edu (Helena Mitasova), vpetras@ncsu.edu (Vaclav Petras), akratoc@ncsu.edu (Anna Petrasova)
URL: baharmon@github.io (Brendan Alexander Harmon)

ϱ = mass of water carried sediment per unit area ($kg \cdot m^{-2}$)

Transport capacity limited landscape evolution

$$\Delta z(x, y, t) = \Delta t \cdot d_s(x, y, t) \cdot \rho_s^{-1} \quad (5)$$

where:
Δz = change in elevation (m)
d_s = net erosion-deposition ($kg\ m^{-2} s^{-1}$)
ρ_s = sediment mass density ($kg\ m^{-3}$)

... [1]

2. Methods

2.1. Implementation

1. Function for sediment flux based landscape evolution
2. Function for erosion-deposition based landscape evolution
3. Function for dynamic modeling based on constant parameters
4. Function for dynamic modeling based on list of rainfall observations
5. Registration in temporal framework
6. Handling of edge effects from moving window computations

This set of python scripts is available on Github at `https://github.com/baharmon/landscape_evolution` released under the GNU General Public License version 2. These scripts are meant to be run inside of GRASS GIS using the GRASS Python Scripting Library. GRASS GIS is an open source project released under the GNU General Public License version 2. GRASS GIS is available at `https://grass.osgeo.org/`.

2.2. Tangible landscape evolution

Tangible Landscape – a tangible user interface tightly integrated with a geographic information system for intuitively sketching in 3D [3]. Conceptually, Tangible Landscape couples a physical model with a digital model in a real-time feedback cycle of 3D scanning, geospatial modeling and simulation, and projection in order to physically manifest digital data as tangible bits. With tangible bits users can directly, physically feel and manipulate data with their bodies – naturally, intuitively understanding space, form, and process. Tangible Landscape is available on Github at `https://github.com/ncsu-osgeorel/grass-tangible-landscape`.

Figure 1: **Rapid prototyping.** 3-axis CNC fabrication of the evolved landscape in polymer-enriched sand using a plunge cut.

We coupled Tangible Landscape with the landscape evolution model to test the model and experiment with strategies for restoration. We used Tangible Landscape to computational steer the landscape evolution model and interactively explore the relationship between overland flow patterns and changes in topography. By manually changing the physical model of the landscape we change the topography used by the model.

3. Results

4. Discussion

4.1. Future work

1. Add water and suspended sediment particles to next run

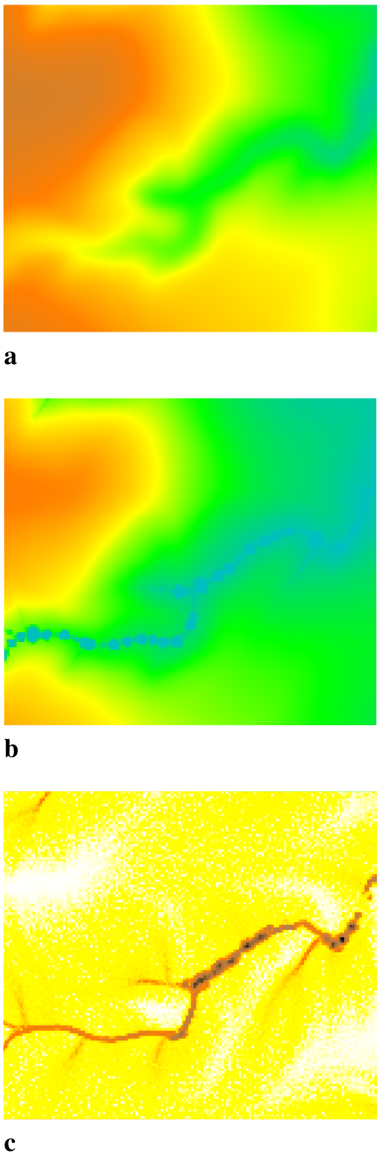


Figure 2: **Sediment flux based gully evolution. a)** A bare earth digital elevation model of gully in Lake Raleigh Woods, North Carolina derived from lidar data. **b)** The simulated evolution of the gully based on a detachment limited soil erosion regime. The landscape evolution model was run as a dynamic simulation with 155 mm/hr rainfall intensity for 5 minutes intervals over a 30 min period. This run of model carved deep pits along the center of the channel. **c)** Simulated sediment flux.

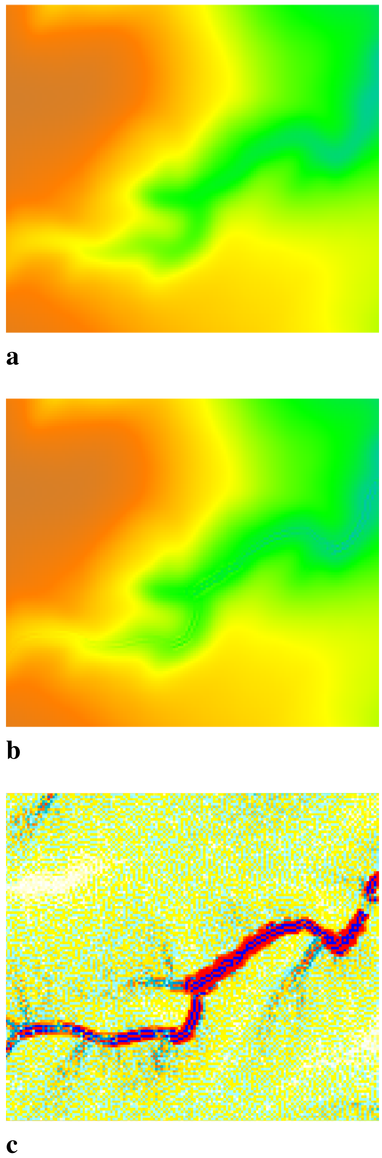
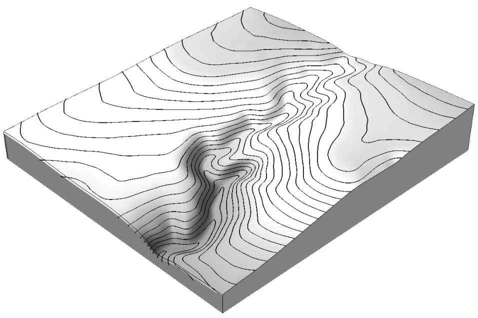
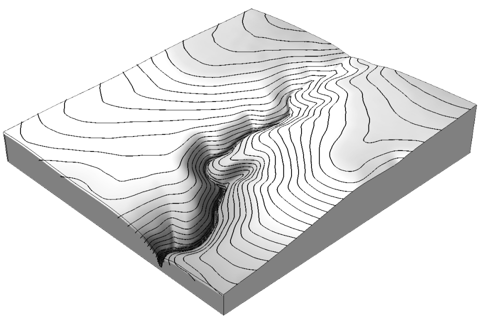


Figure 3: **Erosion - deposition based gully evolution. a)** A bare earth digital elevation model of gully in Lake Raleigh Woods, North Carolina derived from lidar data. **b)** The simulated evolution of the gully based on a transport capacity limited soil erosion regime. The landscape evolution model was run as a dynamic simulation with 155 mm/hr rainfall intensity for 5 minutes intervals over a 30 min period. This run of model carved a deeper channel, accumulated deposited sediment along the centerline of the channel, and accumulated deposited sediments along the banks of the channel. **c)** Simulated erosion-deposition.

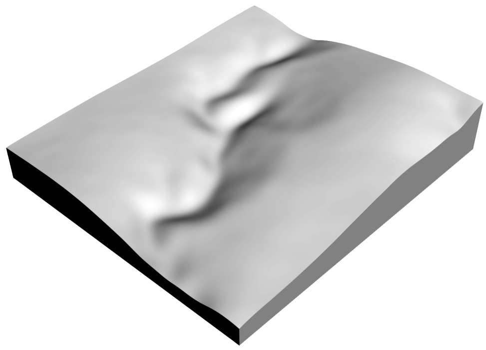


a

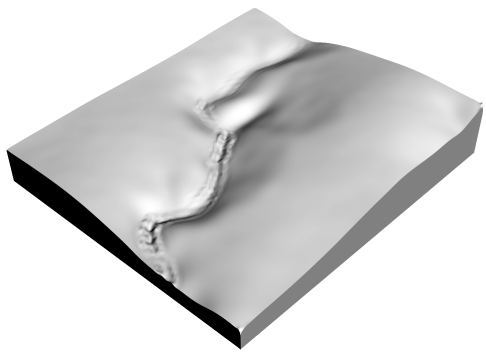


b

Figure 4: **Sediment flux based gully evolution. a)** A gully in Lake Raleigh Woods, North Carolina. **b)** The simulated evolution of the gully based on a detachment limited soil erosion regime. The landscape evolution model was run as a steady state simulation with 155 mm/hr rainfall intensity for 10 minutes to model a 10-year storm event. This run of the model carved a deep incision along the centerline of the channel.



a



b

Figure 5: **Erosion-deposition based gully evolution. a)** A gully in Lake Raleigh Woods, North Carolina. **b)** The simulated evolution of the gully based on a transport capacity limited soil erosion regime. The landscape evolution model was run as a dynamic simulation with 155 mm/hr rainfall intensity for 5 minutes intervals over a 30 min period. This run of model carved a deeper channel, accumulated deposited sediment along the centerline of the channel, and accumulated deposited sediments along the banks of the channel.

2. Test the model on historical data
3. Test the model with field data
4. Empirically calibrate the parameters
5. Refactor code
6. Develop a GRASS GIS addon
7. Implement as a Tangible Landscape analysis
8. Live, in-situ fabrication in polymer-enriched sand with a robotic arm

5. Supporting Information

5.1. S1 File.

Python scripts. (ZIP)

5.2. S2 File.

GIS data. (ZIP)

References

[1] Mitasova, H., Barton, M., Ullah, I., Hofierka, J., Harmon, R., 2013. 3.9 GIS-Based Soil Erosion Modeling. In: Shroder, J. F. (Ed.), Treatise on Geomorphology. Elsevier, San Diego, California, USA, Ch. 3.9, pp. 228–258.
URL `http://www.sciencedirect.com/science/article/pii/B978012374739600052X`

[2] Mitasova, H., Thaxton, C., Hofierka, J., McLaughlin, R., Moore, A., Mitas, L., 2004. Path sampling method for modeling overland water flow, sediment transport, and short term terrain evolution in Open Source GIS. Developments in Water Science 55, 1479–1490.

[3] Petrasova, A., Harmon, B., Petras, V., Mitasova, H., 2015. Tangible Modeling with Open Source GIS. Springer.