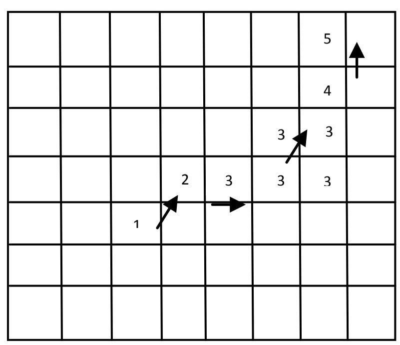
Depiction Runoff Element

Nov 19, 2009

The Runoff element allows simulation of water flow from a point source. The simulation can proceed with a limited supply of water or, alternately, the maximum inundation zone (assuming unlimited supply) can be determined. Flow velocity, general hydrodynamics are not modeled. Soil absorption is modeled indirectly as will become apparent. The basic premise here is that topological data alone, derived from Digital Elevation Models (DEM), contains sufficient information to model watersheds and drainage networks to a fairly high degree of accuracy, with the resolution of the DEM, of course, dictating the end result of the simulation.

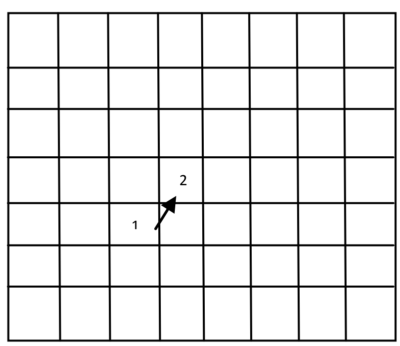
The end result of the simulation is a set of pixels/cells in the elevation grid, each of which maintains a count of the number of pixels that are upstream from it. Assuming that each cell upstream absorbs X gallons of water, a given cell becomes part of the runoff if and when the supply at the source pixel exceeds X times the number of pixels upstream from the given cell. The given cell is then termed 'triggered' and the required volume is the 'trigger' volume for that cell.

Runoff Example

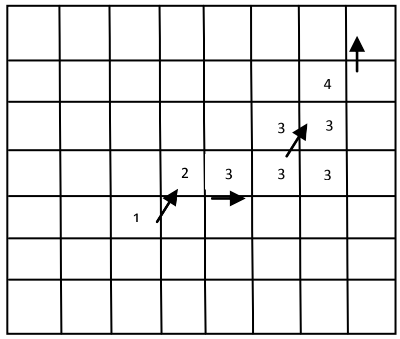


The above example illustrates the runoff process. Based on the topology, it is determined that only the numbered cells are in the potential runoff. The runoff starts at the pixel numbered 1. The pixels numbered 3 are all in a lake with the '4' pixel being the outlet of the lake.

Runoff with 2X gallon supply



Runoff with 8X gallon supply



Notice in the last example that in order for the lake to be included in the runoff, all the pixels in the lake need to be triggered -- i.e., instead of 3X gallons of water, you'd need 8X since the lake itself has 5 cells in it.

<u>The Runoff Algorithm</u>

The algorithm, synthesized broadly from GIS literature, is as follows:

1. Place seed pixel in a **ToBeProcessed** list. **End Condition** for the runoff calculation is reached when this list is empty. Set seed pixel's upstream count to be one.
2. Starting from the seed pixel (location where the Runoff element was placed), until End Condition is reached, determine the lowest neighbor. **Lowest neighbor** determination takes into account the *Height Tolerance* parameter as well as a preferred runoff direction. The latter to prevent unnatural, sudden turns in the runoff.
3. If lowest neighbor is not already in the runoff, add it to the **ToBeProcessed** list. Set its upstream count to be the immediate upstream neighbor's count plus one.
4. If no lowest neighbor can be found, the pixel is in a **Pit**.
5. A **flatness test** is also performed in the neighborhood (fit a plane to the neighborhood and if the plane is inclined less than 0.1 degrees, the neighborhood is deemed flat and considered a Pit).
6. **Flood the Pit** to form a lake. This is how:
    a. Find the neighbor that has the lowest elevation (could be higher than the seed pixel);
    b. Raise the lake to this lowest elevation and expand the shoreline of the lake;
    c. Repeat the 'Flood the Pit' process for all pixels in this newly enlarging lake;
    d. If any neighbor pixel is lower than the current lake level, call that pixel the Outlet and terminate the lake;
    e. If the new lake is adjacent to another lake, merge the lakes;

f. All the pixels in the lake are considered to be part of the runoff;
g. All the pixels in the lake get an upstream count the same as the inlet's upstream account;
h. The lake's outlet pixel's upstream count is the inlet pixel's upstream count plus the sum of all pixels in the lake;
i. If at any point during the flooding of the lake an outlet that's lower than the current outlet is found, then drain the lake to this new, lower outlet.

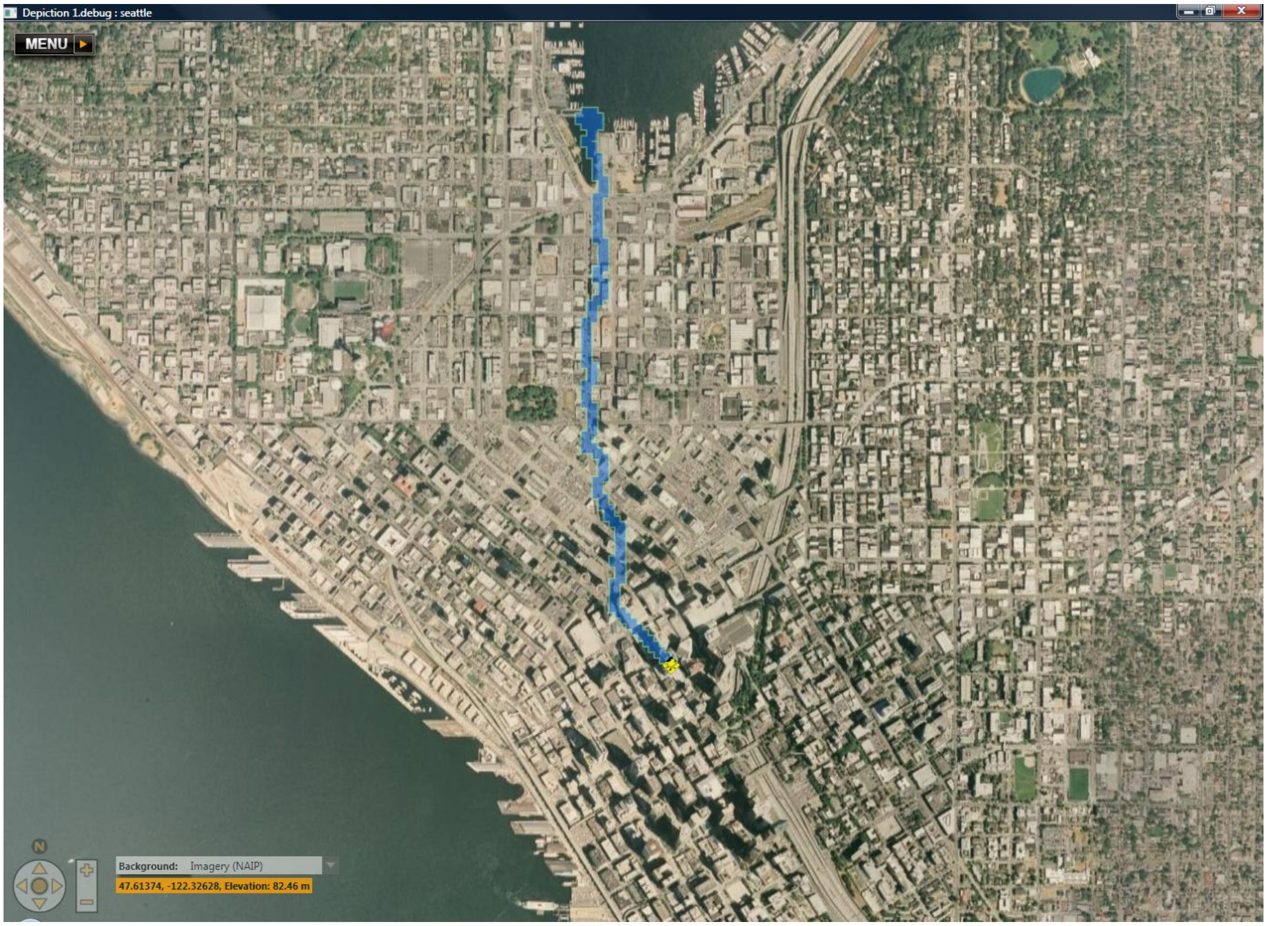
7. When the **ToBeProcessed** list is empty, terminate the runoff calculation.
8. If *Show Maximum Inundation* parameter is set, all the pixels in the runoff are displayed. If not, then the pixels in the runoff are turned on (present in runoff) or off (not present in runoff) based on the number of pixels upstream.

<u>Limitations and additional points to note</u>

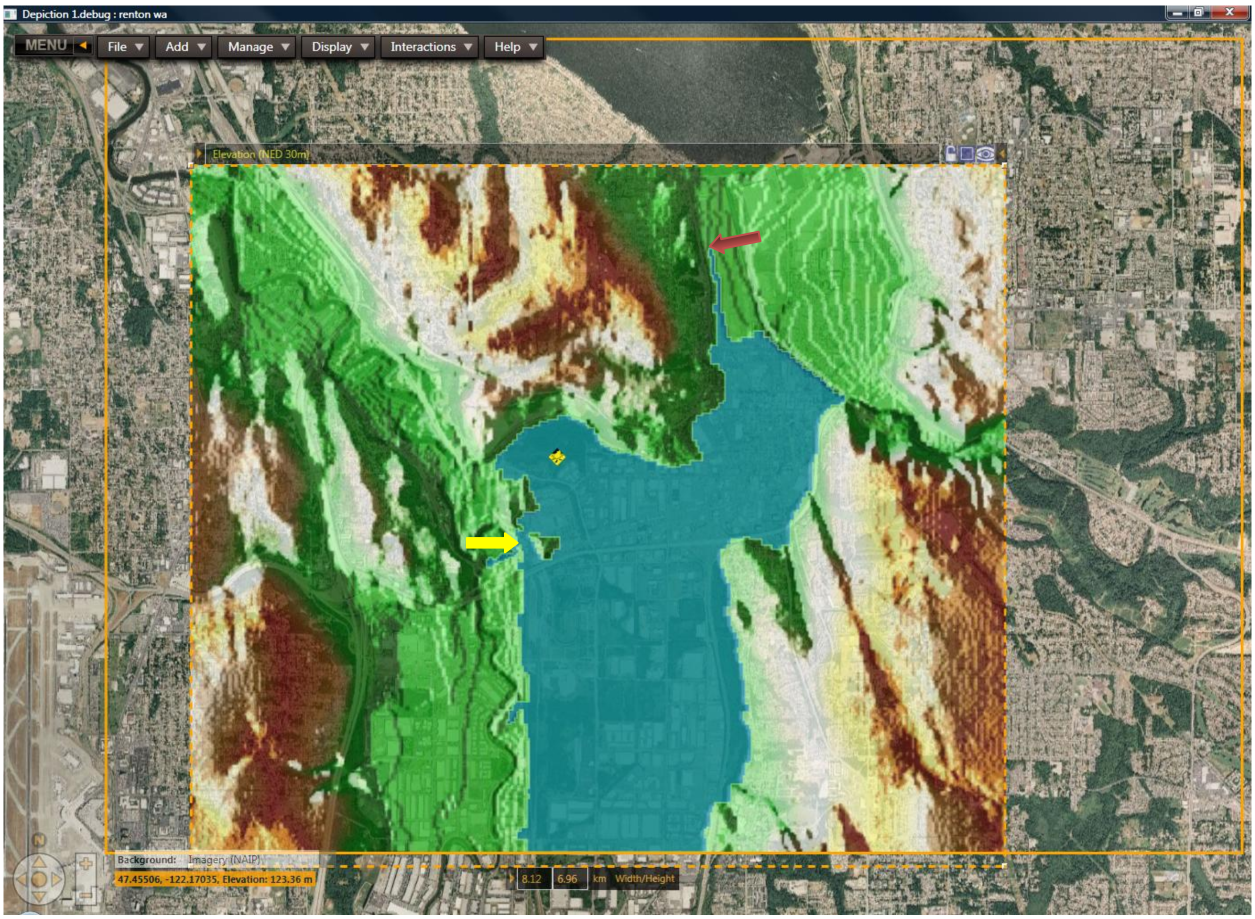
- Unlike natural, real world streams, the runoff this algorithm produces does not bifurcate.
- The absorption as defined by the algorithm is hard-coded to be 0.01m. This means that the trigger volume for each pixel is the number of pixels upstream multiplied by 0.01 * gridPixelArea * 264.17, where **gridPixelArea** is the area in meters squared of an individual DEM pixel (would be 900 for a 30m, NED data downloaded via Depiction quickstart), and the constant 264.17 converts meters cubed to gallons.
- The *Height Tolerance* parameter is used to determine if a neighbor is lower than a center pixel. It is used to compensate for vertical inaccuracies in DEM data. If a center pixel is at 20m and a neighbor is at 20.2m and the *Height Tolerance* is 0.5m, the neighbor is considered to be below the center pixel.
- The *Smoothing Factor* parameter is used to average the DEM data along the horizontal dimension. With allowed values ranging from 0 to 10, square kernels of the specified size (in pixels) are used to average the elevation value at each pixel in the DEM grid.
- Actual drainage networks that are narrower than the resolution of the DEM grid might not be included in the runoff.
- The algorithm performs well at medium (30m) resolution and in highly-sloped terrains and over smaller regions.

Here are a few examples of the Runoff element in action that illustrate the salient aspects, power, and limitations of the simulation model.

Example 1: Reveals the limitation of the lowest-neighbor method of determining runoff. Even though the terrain slopes down westward toward Puget Sound, it slopes down more steeply northward toward Lake Union. So, the runoff takes the longer route to Lake Union.



Example 2: Seed pixel for runoff is in a valley/pit at 8m. Entire valley at 8m elevation is flooded. Even though there is a downward slope in terrain toward Lake Washington (red arrow) starting at 7.6m, the lower outlet is at 5.38m (yellow arrow). The runoff algorithm chooses this lower outlet, but soon runs into a pit it cannot climb out of and terminates.



Example 3: Runoff from Example 2 compared to a Flood Fill (Depiction's Flood element) at zero meters starting at the same location as the runoff. Flood is in red while the runoff is in blue.