Optimized Pit Removal V1.5.1 - Pseudocode

Stephen Jackson

Center for Research in Water Resources

University of Texas at Austin

# Variables

### DATA STORED FOR EACH CELL:

**Terrain value**: Elevation from DEM, modified by algorithm

**Flooded**: Binary state representing rising flood from lowest points on DEM. Flooding is permanent

**Confirmed Descent**: A binary state indicating that there is a known path of equal or descending elevations to an outlet. Confirmed Descent is permanent.

**Flooding Direction**: An indication of which of the eight neighbors caused this cell to be flooded. Used to retrace a path back to an outlet. Flooding Direction is permanent.

**Checked**: Indicates whether a cell has been tested for inclusion in the current depression. Resets after each pit removal.

### PRIORITY QUEUES:

**Main Queue**: A priority queue storing cell location and elevation. Every cell with data will eventually be added and removed from the Main Queue.

**Depression Queue**: A priority queue storing cell location and elevation. Used locally to determine the current depression extent.

### OTHER MAJOR VARIABLES:

**Depression List**: An ordered list of cell locations within the current depression from lowest elevation to highest. Resets after each pit removal.

**Step Size**: A user specified value for the elevation increment for the Cut and Fill Functions. Smaller values yield refined results but greater processing time.

**Mode**: A user specified option for whether Pits should be removed using Cut Only, Balancing Cut and Fill, or Minimizing Absolute Cost.

**No Data**: The value used by the DEM to signify no elevation data is provided.

# DEFINITIONS:

**Outlet**: A cell which is on the border of the grid OR a cell with a neighbor that has No Data.

**Pit**: A cell for which all neighbors are either of higher elevation OR of equal elevation which flood prior and do not have Confirmed Descent. This means for a wide pit with a flat bottom, the last flooded cell (roughly farthest from the outlet) is considered the minimum.

**Neighbor**: One of the eight adjacent cells.

**Crest**: The highest elevation on the path from the current pit to an outlet. The path is traced backwards based on which neighboring cell triggered flooding in the current cell.

**Depression**: A connected group of cells which all drain towards the current Pit. A compound depression (neighboring pits with a separating ridge lower than the crest elevation) is treated as separate depressions.

# FUNCTIONS:

### \*\*\*Main Algorithm\*\*\*

CALL **Get User Settings**

CALL **Import**

CALL **Initialize Main Queue**

CALL **Iterate Main Queue**

CALL **Export**

### \*\*\*Get User Settings\*\*\*

INPUT Original grid DEM file location

INPUT Mode

INPUT Step Size

OUTPUT Pit Removed DEM file location

OUTPUT (Optional) Pit Location file location

### \*\*\*Import\*\*\*

Get value signifying No Data from Original DEM

Get size of Original DEM

FOR each cell in Original DEM

Add value to Terrain

END FOR

### \*\*\*Initialize Main Queue\*\*\*

FOR each cell in Terrain

IF cell is an Outlet

Identify cell as Flooded

Identify cell as Confirmed Descent

Add cell to Main Queue

IF cell is not on border and has neighbor with No Data

Identify cell’s Flooding Direction

END IF

END IF

END FOR

### \*\*\*Iterate Main Queue\*\*\*

WHILE Main Queue is not empty

Remove cell with lowest elevation from Main Queue

Determine if cell is a Pit using CALL **Identify Pit** WITH Pit cell

IF cell is a Pit

CALL **Remove Pit** WITH Pit cell

ELSE IF cell does not have Confirmed Descent but has a neighbor with a lower elevation that has Confirmed Descent

Identify cell as Confirmed Descent

END IF

FOR each neighbor that has not yet been identified as Flooded

Add neighbor to Main Queue

Identify neighbor as Flooded

Identify neighbor’s Flooding Direction

IF cell has Confirmed Descent and neighbor has a higher elevation

Identify neighbor as Confirmed Descent

END IF

END FOR

END WHILE

### \*\*\*Remove Pit\*\*\*

IF selected Mode is cut only

CALL **Cut Routine**

ELSE

CALL **Get Crest Elevation** WITH Pit cell

CALL **Get Depression Extent** WITH Pit cell and Crest elevation

CALL **Create Cut Function** WITH Pit cell and Crest elevation

CALL **Create Fill Function** WITH Pit cell and Crest elevation and Depression List

CALL **Get Ideal Fill Level** WITH Cut Function and Fill Function

CALL **Fill Routine** WITH Pit cell and Ideal Fill Level

CALL **Cut Routine** WITH Pit cell

END IF

Identify Pit cell as Confirmed Descent  
Clear Depression

Clear Cut Function

Clear Fill Function

### \*\*\*Get Crest Elevation\*\*\*

Set current Crest equal to Pit elevation

Start at Pit cell

WHILE cell does not have Confirmed Descent

IF cell elevation is greater than current Crest

Set Crest equal to cell elevation

END IF

Get next cell using CALL **Trace**

END WHILE

### \*\*\*Get Depression Extent\*\*\*

Add Pit cell to Depression Queue

Add Pit cell to Depression List

WHILE Depression Queue is not empty

Remove cell with lowest elevation from Depression Queue

FOR each neighbor

IF neighbor has not been Checked AND neighbor elevation is lower than crest elevation AND neighbor elevation is greater than or equal to cell elevation

Add neighbor to Depression Queue

Add neighbor to Depression List

END IF

Identify neighbor as Checked

END FOR

END WHILE

Clear all Checked

### \*\*\*Create Cut Function\*\*\*

Create discrete Cut Function from Pit elevation to Crest elevation, inclusive, incremented by Step Size

Start at Pit cell

WHILE cell does not have Confirmed Descent

FOR each function elevation in Cut Function

IF cell elevation is greater than function elevation

Add the difference between cell elevation and function elevation to the existing function value

END IF

END FOR

Get next cell using CALL **Trace**

END WHILE

### \*\*\*Create Fill Function\*\*\*

Create discrete Fill Function from Pit elevation to Crest elevation, inclusive, incremented by Step Size

FOR each cell in Depression List

FOR each function elevation in Fill Function

IF cell elevation is less than function elevation

Add the difference between function elevation and cell elevation to the existing function value

END IF

END FOR

END FOR

### \*\*\*Get Ideal Fill Level\*\*\*

*(Note: The current algorithm treats both Cut and Fill as positive quantities. A convention where Cut is negative would be more natural and is recommended for future versions. Balance minimizes net change, while Minimum Cost minimizes net absolute change.)*

IF Mode is Balance

Initialize Minimum Difference as the Cut Function value for the Pit elevation

Initialize Ideal Fill Level as the Pit elevation

FOR each function elevation in Cut/Fill Function

IF the Fill value minus the Cut value is less than the Minimum Difference

Set Minimum Difference equal to Fill value minus Cut value

Set Ideal Fill Level to function elevation

END IF

END FOR

ELSE IF Mode is Minimize Cost

Initialize Minimum Cost as the Cut Function value for the Pit elevation

Initialize Ideal Fill Level as the Pit elevation

FOR each function elevation in Cut/Fill Function

IF the Fill value plus the Cut value is less than the Minimum Cost

Set Minimum Cost equal to Fill value plus Cut value

Set Ideal Fill Level to function elevation

END IF

END FOR

END IF

### \*\*\*Fill Routine\*\*\*

FOR each cell in Depression List

IF cell value is lower than Ideal Fill Level and cell value is not No Data

Set Terrain cell value to Ideal Fill Level

END IF

END FOR

### \*\*\*Cut Routine\*\*\*

*(Note: the Pit cell elevation may have already been modified by the Fill Routine. The Cut Routine acts on the modified value)*

Get Pit elevation

Start at Pit cell

WHILE cell does not have Confirmed Descent

IF cell elevation is greater than Pit elevation

Set Terrain cell value to Pit elevation

END IF

Identify cell as Confirmed Descent

Get next cell using CALL **Trace**

END WHILE

### \*\*\*Trace\*\*\*

Get cell’s Flooding Direction

Return neighbor which caused this cell to Flood

### \*\*\*Detect Pit\*\*\*

Set IsPit equal to True

IF cell does not have Confirmed Descent

FOR each neighbor

IF neighbor elevation is less than cell elevation

Set IsPit equal to False

ELSE IF neighbor elevation is equal to cell elevation and neighbor has not been Flooded

Set IsPit equal to False

END IF

END FOR

ELSE

Set IsPit equal to False

END IF

Return IsPit