Modelling the effect of tillage-induced roughness on runoff patterns

PC-Raster scripts and User Manual

Ingrid Takken

Laboratory for Experimental Geomorphology

K.U.Leuven

Contents

1. Introduction

Oriented roughness on tilled fields often directs runoff along tillage lines instead of in topographic direction (Ludwig et al., 1995; Desmet and Govers, 1997; Souchère et al., 1998; Takken et al., 2001). In addition, borders between fields may act as water collectors and water flow may be routed along linear features, such as lynchets or roads, even if these are oriented more or less parallel to the contour lines. Therefore, the actual runoff pattern can be very different from the runoff pattern that would be predicted from topography alone. Due to the fact that the runoff pattern defines the locations where water will concentrate, as well as the effective slope gradient (ea. the slope in flow direction), this also has an important effect on erosion patterns.

The models described in this manual can be used to create a runoff pattern, taking into account the effects of tillage-induced roughness. The methodology to create the runoff pattern includes four steps. First, a topographically controlled runoff pattern is created using the standard single-flow, steepest descent algorithm. Next, a tillage-determined runoff pattern is created, i.e. a runoff pattern based on the assumption that water will always flow in the tillage direction (TCRP model, Takken et al., 2001a). Then, decision rules are applied to determine for each cell whether the water will flow in tillage direction or in topographic direction (Takken et al., 2001b). Finally, a flow direction map can be generated by combining the topographically determined runoff pattern with the tillage-controlled runoff pattern (Takken et al., 2001b).

The procedures in the models are described in more detail by Takken et al., 2001a and b. In this manual the use of the model scripts is explained. As the correction methods applied in the TCRP model are not described in the papers of Takken et al., some explanation on these correction methods is given in the appendix.

The model scripts are written in PCRaster, which is a computer language incorporated in a GIS (Van Deursen, 1995). To understand and use to model scripts, PCRaster version 4.4.0 is required. Installation of PCRaster can be done using miniconda, see for details: https://pcraster.geo.uu.nl/pcraster/4.3.3/documentation/pcraster_project/install.html.
Nutshell is a graphical user interface for PCRaster on windows and can be downloaded from: https://sourceforge.net/projects/nutshellqt/files/latest/download. Input and output maps are in PCRaster format. The user is referred to the PCRaster manual for explanation on PCRaster functions, map types etc. (Karssenberg, 1996), newest version available on: https://pcraster.geo.uu.nl/downloads/latest-release/.

2. The topographic runoff pattern

To calculate the topographic runoff pattern standard PCRaster functions can be used. It is recommended to remove pits from the DEM first by typing:

```
pcrcalc Dem.map = lddcreatedem(Dem.map, 1e35, 1e35, 1e35, 1e35)
```

Subsequently the topographic local drain direction map is calculated by :

pcrcalc LddTop.map = lddcreate(Dem.map, 1e35, 1e35, 1e35, 1e35)

In addition the slope (in m/m) and the aspect directions can be calculated by PCRaster procedures:

pcrcalc Slope.map = slope(Dem.map)
pcrcalc Aspect.map = aspect(Dem.map)

3. Application of the TCRP model.

The TCRP model creates a runoff pattern (Local drain direction map) based on the assumption that water will always flow in tillage direction (Takken et al., 2001a).

3.1 Scripts

In order to create this runoff pattern several model scripts need to be run after each other. All model scripts are text files that can be opened in any editor.
The model scripts for the TCRP model have the following names:
LddTill.mod (or LddTill1.mod)
PitRem.mod
CirkRem.mod
KruisRem.mod
Kruis.mod

3.2 Input

The following input maps are required for the TCRP model:

Dem.map: A digital elevation model that is larger than the catchment area (scalar).

Fields.map: A map with fields (nominal). Also pastures and forests need to have a separate fields-identifier. The map needs to be larger than the catchment area and the numbers 0 and 1 should be avoided on the Fields.map.

TillDir.map: A map with tillage orientations (0-180°). The map needs to be of the same size as the fields map, with –1 for non-tilled areas. The directions on this map (ea. PC-Raster 'directional' map type) is zero to the north and positive in clockwise direction. The data type of this map should be scalar.

Chan.map: A map with the location of channels and roads that act like channels, such as sunken lanes Boolean: 1 for channels, 0 for the rest. (If there are no channels this map should only contain zeros).

ChanLdd.map: A map with the flow directions in the channels. (Ldd for channels, missing values for the rest. If there are no channels, this map should only contain missing values.)

3.3 Procedure

Firstly, the "LddTill.mod" or "LddTill1.mod" script is applied. This is the main part of the TCRP model and it creates a tillage-controlled runoff pattern. If "LddTill1.mod" is used , the runoff pattern is calculated according to method 1, described by Takken et al. (2001a). Hereby, for each cell the flow is directed to the neighbour cell in the direction that corresponds best with the tillage direction. If "LddTill.mod" is used, method 2 (Takken et al., 2001a) is applied. Hereby, tillage lines are created that correspond better to the tillage direction over longer distances. Although, it may be interesting to use method 1 (LddTill1.mod) for comparison, method two will give the best results and is therefore recommended.

The model scripts can be opened in an editor. Under the 'binding' section the names of input and output maps can be changed. (Default names are given above).

The model can be run by the following command:

```
pcrcalc –f LddTill.mod
```

The model will give the following **output** maps:

<u>LddTill.map</u>: The calculated tillage-controlled runoff pattern. (This map is called "LddTill1.map", if the "LddTill1.mod" script is used).

<u>SlopeTil.map</u>: The slope (m/m) in direction of flow (note: for the use of LISEM the slope may not be smaller than 0.001). (The map is called "SlopeTi1.map", if "LddTill1.mod" is used).

<u>TillDir2.map</u>: Directional map with tillage directions, including headlands.

<u>FlowDir.map</u>: Directional map containing the flow directions in degrees, i.e. not the Ldd directions, but aspect or tillage directions. (This map is required if the unit contributing area has to be calculated).

<u>Angle.map</u>: A scalar map with the angle between the topographic and tillage direction (in degrees). This map is required in the model script (Logit.mod) to predict flow directions.

<u>MapEdge.map</u>: Boolean map showing the edge of the Fields.map. This map is required for the pit-remove programs.

If there are flat areas in the DEM it is possible that pits occur in the calculated local drain direction (Ldd) map. To check if the Ldd-map contains pits, the PC-Raster command 'pit' can be used. Type:

```
pcrcalc Pits.map = pit(LddTill.map)
```

and then :

```
aguila Pits.map
```

If this map gives only 0 values, the "LddTill.map" is ok! If pits are found (i.e. values > 0), the following pit-remove programs need to be applied.

There are two types of pits, ea. pits caused by "loose-ends" and pits caused by "circular flow". The model scripts "PitRem.mod" and "CirkRem.mod" can be applied to remove the two types of pits respectively. For a description of the correction methods see the appendix.

Firstly apply the "PitRem.mod" script by typing :

pcrcalc –f PitRem.mod

(The default input name of the Ldd-map is "LddTill.map, if your map has a different name, the PitRem.mod file needs to be edited to give the correct names under 'binding'.)

This model creates an output map (Ldd data type) named "PitRem.map".

Check again if this map still contains pits : "pcrcalc pits = pit(PitRem.map)".
If this map is ok., it can be used as input to "KruisRem.mod", otherwise "CirkRem.mod" has to be applied first, by typing:

pcrcalc –f CirkRem.mod

This gives an output map (Ldd data type): "CirkRem.map".

This map should not contain any pits anymore. If there are still pits, repeating the "PitRem" and "CrikRem" models may help.

Finally, "KruisRem.mod" should be applied. This removes crossing flow lines, that may have been caused by the changes made in "PitRem" and "CirkRem". However, even if there are no crossing flow lines, the application of this script is useful, because it also re-calculates the slopes in direction of tillage for the corrected Ldd-map.
Edit "KruisRem.mod" in order to give the name of the Ldd.map that need to be corrected, under "INPUT" in the "binding" section.
Then run the "KruisRem" model by typing:

pcrcalc –f KruisRem.mod

This should give you the final tillage-controlled Ldd-map, which is called "LddCor.map" (if this was not changed as output file name under 'binding').

The model script "Kruis.mod" can be used to identify if a Ldd-map contains crossing flow lines. The model needs to be edited to give the name of the file that needs to be checked as input map (under 'binding'). Then type: pcrcalc –f Kruis.mod. This model only gives a map, with the location of crossing flow lines, it does not correct the Ldd map!

3.4 Note

The models need to be applied in the right order. If "CirkRem.mod" is applied, before "PitRem.mod" too many cells may get changed! The KruisRem.mod model always needs to be applied, if PitRem.mod (and CirkRem.mod) are applied, even if the calculated Ldd map does not contain crossing flow lines, because the slope in direction of flow is also calculated in this script.

3.5 Output

The final output of the "KruisRem.mod" script is :

<u>LddCor.map</u>:	The corrected tillage-controlled runoff pattern.
<u>SlopeCor.map</u>:	The slope (m/m) in direction of flow (according to the flow directions on LddCor.map).
<u>FlowdCor.map</u>:	Directional map containing the corrected flow directions (degrees).

4. Application of the model to predict flow directions.

Takken at al., 2001b developed logistic regression equations to predict flow directions on tilled fields. In this model the equation is applied that makes use of the topographic slope, the angle between tillage and aspect direction and the degree of oriented roughness.

4.1 Script

The model script to calculate the flow direction on tilled fields is named:
Logit.mod

4.2 Input

<u>Dem.map</u>:	A digital elevation model (scalar).
<u>Fields.map</u>:	Map with fields (nominal).
<u>Tilldir2.map</u>:	A map with tillage directions (directional), including headlands. This map is made by the TCRP model (LddTill.mod).
<u>Angle.map</u>:	Map with the angle between the tillage orientation and the aspect direction. This map is created by the TCRP model (LddTill.mod).
<u>Roughnes.map</u>:	A map with for each field the order of oriented roughness in cm (scalar).

4.3 Procedure

The model can be run by typing:

```
pcrcalc –f Logit.mod
```

4.4 Output

The model generates two output maps :

<u>Logit.map</u>:	Map showing the logit of the probability that water will flow in topographic direction (scalar).
<u>ToporTil.map</u>:	Boolean map showing whether flow will be in topographic (0) or in tillage (1) direction.

5. Combination of the topographic and tillage-controlled runoff patterns.

Finally, a realistic runoff pattern can be created by combining the topographic runoff pattern (LddTop.map) with the tillage-controlled runoff pattern (LddCor.map) using the ToporTil.map created by Logit.mod.

5.1 Script

The model script that can be used to do this is: LddRes.mod

5.2 Input

Dem.map:	Digital elevation model (scalar).
Chan.map:	Map with the location of channels and roads that act like channels, such as sunken lanes (Boolean : 1 for channels, 0 for the rest. If there are no channels this map should only contain zeros).
LddCor.map:	Map with the corrected tillage-controlled runoff pattern (output of the TCRP model).
ToporTil.map:	Boolean map showing whether flow will be in topographic (0) or in tillage (1) direction (output of Logit.mod).
SlopeCor.map:	Map with slopes in direction of flow according to the LddCor.map (output of the TCRP model). This map is required in order to calculate the slopes in direction of flow for the resulting runoff pattern.

5.3 Procedure

The model can be run by typing:

pcrcalc –f LddRes.mod

5.4 Output

The model generates two output maps :

LddRes.map:	The resulting runoff pattern.
SlopeRes.map:	Map with slopes in direction of flow.

5.5 Note

The model script (LddRes.mod) includes a procedure to correct crossing flow lines that may have come into existence after the combination of the two ldd-maps. However, it is required to check whether the resulting LddRes.map has new pits, by typing:
"pcrcalc Pits.map = pits(LddRes.map)" and then "aguila Pits.map".

If this map gives only 0 values, the "LddRes.map" is ok! If pits are found (values > 0), again the pit-remove programs need to be applied. Hereto, the default name of the input map "LddTill.map" in the "binding" section of the "PitRem.mod" should be changed to "LddRes.map". The name of the output map in the "binding" section of "KruisRem.mod" should also be changed to a new name. (e.g. "LddResC.map"). Then apply again subsequently:

pcrcalc –f PitRem.mod
pcrcalc –f CirkRem.mod

pcrcalc –f KruisRem.mod

6. Additional model scripts.

6.1 General map information.

The model script "Info.mod" calculates some basic map information. In the binding section the name of the map of which the information is required needs to be given. The model then calculates the number of rows and columns and gives the minimum and maximum x and y co-ordinates. To run the model type:

pcrcalc –f Info.mod

Although the output is actually non-spatial, it is given in the form of a number of output maps:

rows: The number of rows
cols: The number of columns
xmin: The minimum x-co-ordinate
xmax: The maximum x-co-ordinate
ymin: The minimum y-co-ordinate
ymax: The maximum y-co-ordinate

6.2 Calculation of the catchment and (unit) contributing area.

In the script "Catch.mod" the catchment area is calculated for a certain outlet point, using a specific Ldd-map. Also the contributing and unit contributing areas are calculated for each grid cell in the catchment.

Input

Ldd.map:	An Ldd-Map ("LddCor.map" or "LddTop.map")
Outlet.map:	Boolean map with 1 for the outlet and 0 for the rest
FlowDir.map:	A Directional map with flow directions, i.e. "FlowdCor.map" if "LddCor" is used and "Aspect.map" if "LddTop" is used.
Slope.map:	A slope map : i.e. "SlopeCor.map" if "LddCor" is used and "Slope.map" if "LddTop" is used.

The model is run by typing:
pcrcalc -f Catch.mod

Output

Catch.map:	Boolean map with the catchment area.
ContArea.map:	Scalar map with the contributing area for each grid cell
UnitCA.map:	Scalar map with unit contributing areas

7. Batch file

If the (default) input file names are used as described above, the model scripts to calculate the tillage-controlled and combined runoff patterns can be run directly after each other using the batch-file : "go.bat". The input maps required to run the go.bat file are : Dem.map, Fields.map, TillDir.map, Chan.map, ChanLdd.map and Roughnes.map (see descriptions above). Note: the models as well as the input files should be stored in the same map directory.

This file can then be run by double clicking it or by typing "go" after the Dos-prompt.
The batch file contains the following lines:

```
pcrcalc -f LddTill.mod
pcrcalc -f PitRem.mod
pcrcalc -f CirkRem.mod
pcrcalc -f KruisRem.mod
pcrcalc -f Logit.mod
pcrcalc -f LddRes.mod
```

References

Karssenberg, D., 1996. PC-Raster, version 2, manual. Department of Physical Geography, Utrecht University, The Netherlands, 383 pp. http://www.pcraster.nl.

Takken, I. , Jetten, V., Govers, G., Nachtergaele, J. and Steegen, A., 2001a. The effect of tillage-induced roughness on runoff and erosion patterns. Geomorphology, 37, in press.

Takken, I., Govers, G., Steegen, A., Nachtergaele, J. and Guérif, J., 2001b. The prediction of runoff flow directions on tilled fields. Accepted for publication in Journal of Hydrology.

Van Deursen, W.P.A., 1995. Geographical information systems and dynamic models. Development and application of a prototype spatial modelling language. Ph.D. thesis, University of Utrecht, The Netherlands, NGS 190, 1-198 pp.

Appendix:

Description of the corrections made for crossing flow lines and pits.

1. Corrections on crossing flow lines

There are situations in which the flow directions on the created map may cross each other diagonally. Firstly, these cells are identified by checking for each cell with a diagonal drain direction, if one of the cardinal neighbours has a flow direction that will cross the flow direction of the cell. If the cell in **anti-clockwise** direction from the flow direction causes a crossing flow line, **error1** is assigned, if the cardinal neighbour in **clockwise** direction from the flow direction causes crossing flow the cell is assigned **error2**. If crossing flow occurs, there will always be a cell of error1 and of error2 next to each other (Figure 1).

To be able to make corrections we need to define first which of the two cells needs to be corrected. Hereto, it was necessary to analyse firstly in which situations crossing flow lines may occur. There are three situations that may cause crossing flow lines:

A. A small channel within a field, that is oriented diagonally over the grid pattern and with a flow direction perpendicular to the flow direction of the field (forest or pasture) (Figure 2a).
B. If there is a thalweg within a field, which is oriented diagonally over the grid pattern and perpendicular to the tillage direction (Figure 2b).
C. If a parcel border is oriented diagonally over the grid pattern and the flow direction of the upper field (forest or pasture) is perpendicular to the flow direction along the parcel border (Figure 2c).

These cells are localised according to the following criteria:
Localisation criteria for case A:
- Cell has error1 or error2
- The downstream cell has the same field identifier as the cell itself
- The two cardinal neighbours of the cell in directions closest to the flow direction in the cell have the field identifier of a channel

Localisation criteria for case B:
- Cell has error1 or error2
- The neighbour cell that has the crossing flow direction was a pit before, replaced by topographic direction

Localisation criteria for case C:
- Cell has error1 or error2
- The downstream cell has another field identifier than the cell itself (border is crossed)

If the cells that need to be corrected are identified, the corrections are made by turning the direction +45° in case of error1 and -45° in case of error2 (Figures 1 and 2).

After these corrections is checked if there are still any crossing flow lines left. (This hardly ever happens). If so, It may be caused by a very small tilled field within another field. In this case it is difficult to define which of the two cells should be given priority and both cells are corrected.

2. Removal of pits

The created tillage-controlled runoff pattern (LddTill.map) may contain pits (cells without outflow direction). This mainly occurs on flat areas of the DEM. They can be found with the 'lddrepair' function of PC-Raster. This function corrects a local drain direction map (Ldd-map) and assigns pits to 'loose ends' and on circular flow-lines. The location where the pit on a circular flow line is created is random. Replacing these pits on the LddTill map by replacing them with the topographic flow direction will often not solve the problem. There are two situations in which this happens:

1. The topographic direction of the pit-cell points to a cell that drains to the pit (Figure 3).
2. The cell on a circular flow line that was assigned a pit, already had a flow direction equal to the direction of the topographic drainage network (Figure 4).

2.1 Model to remove pits caused by 'loose ends'

Firstly all pits on the map (except the ones at the edges of the map sheet) are replaced by the topographic flow direction. Then the flow direction of the cell in downstream direction of the former pit is checked: In case that this direction is back to the direction of the former pit, this cell is made a pit (in stead of the former pit itself). After that ‘lddrepair’ is applied and will not define another pit in the same cell again. This procedure is then repeated several times, which will solve pits of type 1 (Figure 3).

2.2 Model to remove pits caused by circular flow

Pits of type 2 are not solved in the procedure above, because the cell in downstream direction of the former pit, does not drain to the former pit and is therefore not recognised as new pit. Lddrepair will create the same pit again. Pits of type 1 are already removed from the drain direction network, thus the remaining pits result from circular flow. Starting at the location of these pits, a (Boolean) map is created that shows all cells that are part of a circular flow line (CIRCLE). To do this the pit is defined as circle cell first, then the cell upstream of the pit is made a circle cell and then the cell upstream of that cell is made a circle cell and so on. This is done until all cells in the circle are defined as circle cells. (It is possible that a cell has more upstream cells. Then they are all identified as circle cells, but that does not cause any problems later). (In fact this ‘find circle’ procedure is repeated many times, because PC-Raster is not able to do a ‘do …until’ loop).

Once the CIRCLE map has been created, the cells around the circle (cardinal neighbours) are identified as circle neighbours. Amongst these neighbour cells the lowest neighbour is identified. (If there are more cells with the 'lowest' elevation amongst the neighbour cells, more ‘lowest’ neighbours are defined). Then the flow direction of all circle cells that have a ‘lowest neighbour’ is changed to the topographic direction.

It is possible that the circular flow was part of an isolated part of the ldd and that this solution to circular flow has lead to a new ‘loose end’. Therefore the first pit remove procedure is repeated after the circle remove procedure.

Figure captions

For figures see PowerPoint file: Correction_fig.ppt

Figure 1. Errors 1 and 2 are defined for two adjacent cells with diagonal flow directions that cross each other. Either error 1 or error 2 needs to be corrected.

Figure 2. Examples of situations, where crossing flow lines may occur and these situations after corrections are made. A. Flow over a small channel within a field. B. Flow within a thalweg. C. Flow over a field border.

Figure 3. Situation with pits at 'loose ends' (that can not be solved by replacing pits with topographic flow direction) and the method to correct these pits.

Figure 4. Situation with pit due to circular flow and method to correct these type of pits.