The WDPM User's Guide

Version 2.0

Kevin Shook, Robert Armstrong, Oluwaseun Sharomi,
Ray Spiteri, John Pomeroy, Tonghe Liu

January 31, 2021

Contents

1 Introduction

1.1 What is the WDPM?

The Wetland DEM Ponding Model (WDPM) was developed by the Centre for Hydrology at the University of Saskatchewan to model the distribution of runoff water on the Canadian Prairies. The program was originally written in Fortran 95 by Kevin Shook in 2008. Because the program requires many thousands of iterations to converge, it was adapted to CPU parallel processing in 2010 using the OpenMP API (http://openmp.org/). This version was used operationally by the LIRA project, as described in Section 4 on page 31.

In version 1.0, the program was made faster and easier to use. The WDPM code was ported from Fortran to C, and a graphical user interface was added by Oluwaseun Sharomi and Ray Spiteri of the Department of Computer Science at the University of Saskatchewan. At the same time, the parallel processing was changed to OpenCL (http://www.khronos.org/opencl/)), which has been found to be faster, and which supports the use of Graphical Processing Units (GPUs). Funding for the recoding of the WDPM was provided by Agriculture and Agri-Food Canada (AAFC).

In version 2.0, the GUI code was converted from Python 2 to Python 3 by Tongue Liu, and Ray Spiteri of the Department of Computer Science at the University of Saskatchewan. Many bugs were fixed, and the program execution speed was improved by adding a user-selectable water zero threshold depth, as described in section 1.11.

1.2 Licence

The WDPM is distributed under the GPL version 3. The licence is listed in full in Section 5 on page 39.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

1.3 What does the WDPM do?

The purpose of the WDPM is to model the spatial distribution of runoff water on the Canadian Prairies. Because of its recent post-glacial history, the Canadian Prairies do not have a conventional drainage system. When excess water runs off the landscape, generally due to snow melt in the spring, it may trapped in surface depressions ranging in size from puddles to permanent wetlands, and may cause local flooding. If the depressions are full of water, they may connect.

The WDPM was developed to model the spatial distribution of water over a Prairie landscape, as represented by a Digital Elevation Model (DEM). Originally, the purpose of the program was to determine the fractions of Prairie basins contributing flows to streams, as these change dynamically with the storage of water in the depressions (Shook and Pomeroy, 2011; Shook et al., 2013). However, the model has also been used to demonstrate the extent of flooding on Prairie landscapes, which may be useful for operational purposes.

1.4 Program support

No support is provided for the use of this program. If you discover bugs, please contact the authors though the GitHub repository described on page on page 19.

1.5 Citing the WDPM

A paper describing the WDPM is currently being submitted to the Journal of Open Source Software. Until this paper is published, please cite Shook et al. (2013) to refer to use of the WDPM in any publications.

1.6 Program limitations

The WDPM is NOT a hydrological model. It does NOT determine the depths of water applied to or removed from the DEM. The Centre for Hydrology at the University of Saskatchewan has developed the Cold Regions Hydrological modelling (CRHM) platform (Pomeroy et al., 2007) which is capable of modelling the unique hydrological processes of the Canadian Prairies. It is possible to use CRHM's modeled runoff and evaporation fluxes as inputs to the WDPM.

The WDPM is NOT a hydraulic model. It CANNOT be used to model the rate of flow in any type of channel. It does not have a time step. FLUXOS-OVERFLOW is a 2-D hydrodynamic model capable of modelling overland flow rates in the Canadian Prairies (Costa et al., 2020) but is much slower than WDPM to execute and requires more forcing data.

The WDPM requires a DEM to execute, and all DEMs are imperfect representations of reality. Some of the problems with using common DEMs with the WDPM in the Prairies are discussed in Section 1.12 on page 13. It is very important to note that the WDPM only operates at elevations *above* the elevations of the water present when the DEM was constructed.

In the Canadian Prairies, the road network prevents the WDPM from accurately distributing runoff. To allow the water to be distributed properly, it is necessary to breach the roads in the DEM where there are culverts and bridges (Lindsay, J.B., 2016).

1.7 Model inputs and outputs

All the DEM and water data file inputs and outputs are ArcGIS ASCII (.asc) files. It is assumed that the user has access to a geographical information system (GIS) program. For more information, see Section 3.5.3 on page 30. The files used as inputs and/or outputs are listed in Table 1.1.

Table 1.1: Files used as inputs and outputs for the WDPM.

Name	Description	Input/Output	Mandatory?
DEM	Surface elevations	Input	Yes
Water	Water depths	Input	No
Output	Water depths	Output	Yes
Scratch	Water depths	Output	No
`temp.asc`	Water depths	Both	No
`report.txt`	Program messages	Output	No
`.png` file	Water image	Output	No
`colormap_black.txt`	Color map	Input	No

The Water file is specified to change an existing set of water elevations, i.e. to add more water, remove water, or to drain the water. If the DEM initially contains no water, then the file name is set to NULL. Note that the water file name MUST be specified for the Subtract or Drain modules to be used.

The Output file contains the depths of water after the execution of the program. Once the file has been created, it can be used in another run, by specifying it as the Water file.

The Scratch file is similar to the output file, except that it is output every 1000 iterations, and it overwrites the previous version. The purpose of the Scratch file is to allow you to stop the program and to re-start it again. In this case, you would specify the previous Scratch file as the Water file in the new run. If the Scratch file is not used, the file name is set to NULL.

The file `temp.asc` is created automatically, and is used to improve performance. The file is only created after the first 1000 iterations. It is automatically deleted after the end of the run.

In addition to the .asc files, all model messages that are written to the screen are also logged to the file `report.txt`. The model is also capable of reading the run parameters from a text file as described in Section 3.2.4 on page 25.

The WDPM can create a simple image of the water output, using the open-source program **gdaldem**, which is part of the GDAL suite of programs (GDAL/OGR contributors, 2020). This is not intended to be a substitute for a GIS program, but does provide the ability to create an image of the water surface as a .png file. The program requires a file called `colormap_black.txt` which must be in the same directory as the WDPM executable file. The structure of the colour map file is explained in Section 3.5.3 on page 30.

1.8 Modules

The WDPM applies simulated water to a digital elevation model of a Prairie basin using three modules: Add, Subtract and Drain.

1.8.1 Add

This module adds a specified depth of water to the basin. If the DEM is dry prior to the addition of water, a file created containing the water depths for each cell of the DEM. If there is an existing water file, then the specified depth of water is added to the existing water, and the total is redistributed. This module is intended to (roughly) simulate the addition of excess water to depressional storage by runoff from snowmelt or intense precipitation.

The addition of water can be slow, particularly if a very fine tolerance is used. If the basin is to be drained, then it is not necessary to use a fine tolerance, as the water will be redistributed while being drained.

If a stream exists in the DEM, then the Add module will route water to the stream channel. However, because of the way that the algorithm works, the edges of the

DEM acts as dams, preventing any the water from leaving the DEM. This causes the modeled stream to 'back up' over the landscape as shown in Figure 1.1.

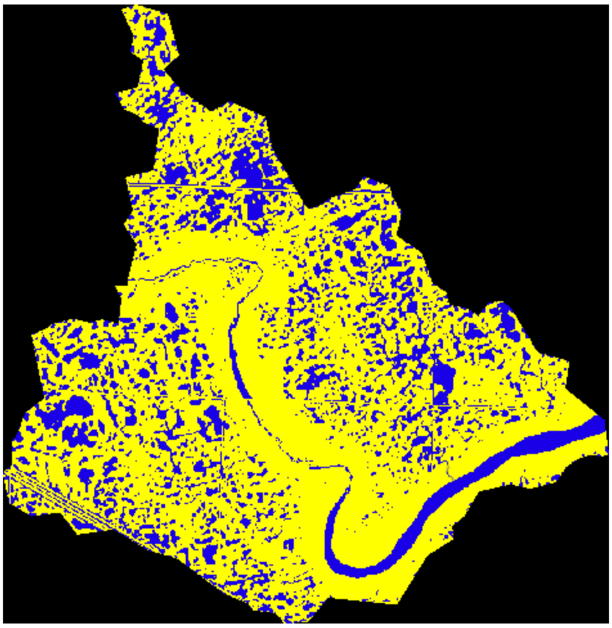

Figure 1.1: 100mm of water added.

To simulate the effects of infiltration of precipitation/snowmelt to soil, the addition of water allows the use of a runoff fraction (0-1) on the water that is applied to directly to land. The runoff fraction is not used on the water applied to existing water.

1.8.2 Subtract

This module subtracts a specified depth of water from each DEM cell to represent evaporation. No spatial variability is in the evaporation is currently allowed. This module generally executes very quickly as very little spatial redistribution of water is usually required. Figure 1.2 shows the removal of 50 mm of water from the water file created from the addition of 100 mm.

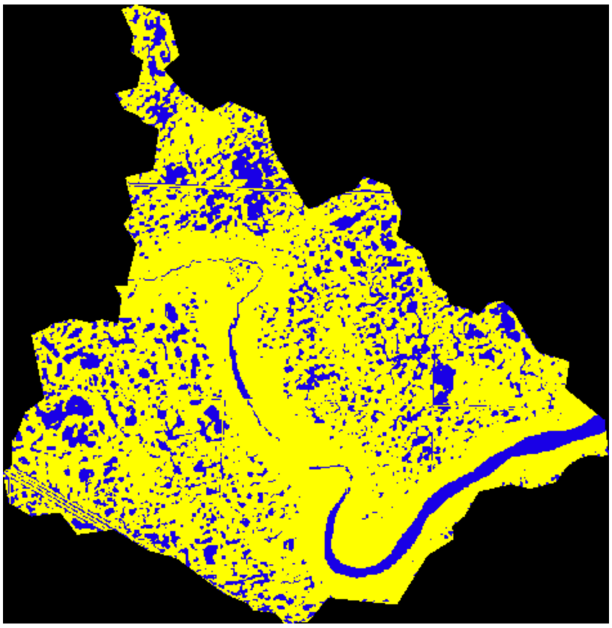

Figure 1.2: 100mm of water added, 50mm removed.

1.8.3 Drain

This module drains the water on the DEM through the lowest point, assuming that this point is in the drainage system. This module can be the slowest to execute (depending on the resolution and the convergence parameters), as it moves large volumes of water long distances. The purpose of the Drain module is to eliminate the 'backing up' of water over the DEM. Figure 1.3 on the next page shows the draining of the water resulting from the addition of 100 mm, as shown in Figure 1.1 on the preceding page. Note that the stream is now essentially dry. If your DEM does not include a stream (as is true of many landscapes on the Canadian Prairies, then you may not need to run the Drain module.

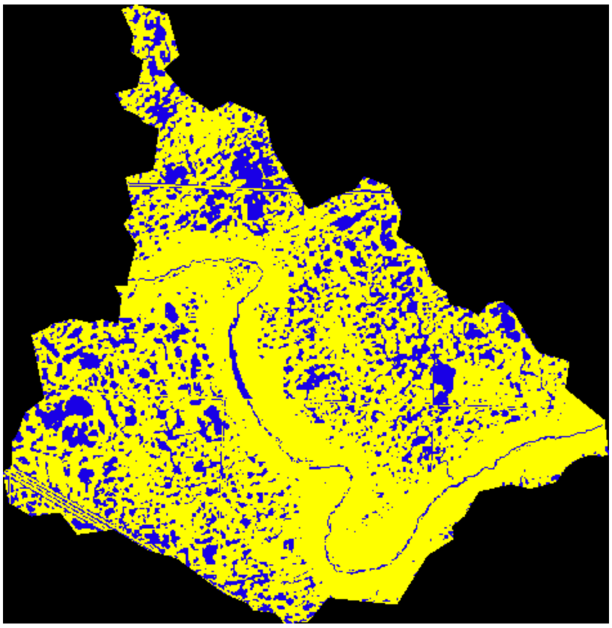

Figure 1.3: 100mm of water applied, then drained

The modules may be executed in varying order. For example, it may be desired to add water, to simulate spring runoff, followed by draining, and then to remove water to simulate evaporation. This could be followed by the addition of water to simulate summer rainfall. As was demonstrated by Shook et al. (2013), the addition and removal of water are non-reversible. Each process affects the spatial distribution of water differently.

1.9 Algorithm

The redistribution algorithm of the WDPM, which is used by all three modules, is taken from Shapiro and Westervelt (1992). The algorithm and its implementation in the WDPM are described in Shook and Pomeroy (2011) and Shook et al. (2013). Unlike the D8 direction of drainage algorithm used by programs such as TOPAZ Garbrecht and Martz (1997), the Shipiro and Westervelt (SW) algorithm allows drainage in more than one direction, as shown in Figure 1.4. Most importantly, the SW algorithm actually moves simulated water over the landscape. When water is added, it runs into surface depressions. When water is removed, the water levels in the depressions are reduced. Therefore, the WDPM does not require the DEM to be processed to remove pits before it is used. It also means that the landscape drainage changes dynamically, as water is added or removed.

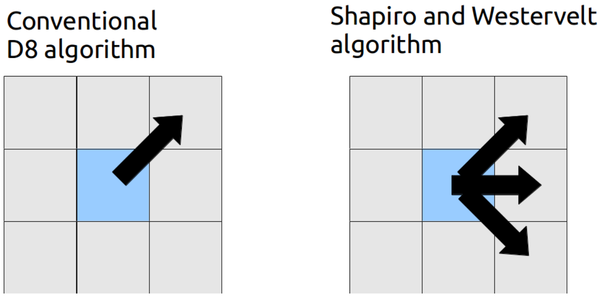

Figure 1.4: D8 and Shapiro and Westervelt algorithms

Unlike D8 drainage, the SW algorithm is iterative. The algorithm is applied to each element in the DEM as shown in the schematic diagram in Figure 1.5. Each is compared separately to its 8 neighbours, looking at the water surface elevation.

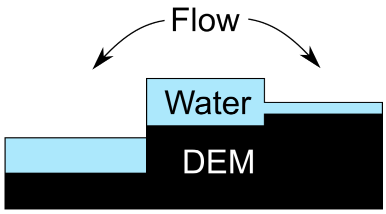

Figure 1.5: Schematic side view of the Shapiro and Westervelt water distribution algorithm.

Each iteration consists of selecting each cell in the DEM, one at a time, and applying Algorithm 1.1 to the selected cell and its neighbours.

Algorithm 1.1 Pseudocode for Shapiro and Westervelt algorithm as implemented by the WDPM.

```
if water surface elevation of selected cell > that of the neighbour cell
  if selected cell DEM elevation > neighbour cell DEM then
    move one eighth of the water in the selected cell to the neighbour cell
  else
    move one eighth of the difference in water elevation to the neighbour cell
  end if
end if
```

At each iteration the algorithm is imperfect as the depth of water transferred may be result in an inaccurate representation of the final water surface. However, over many thousands of iterations, the movement of water will result in a realistic water surface, where the local water elevation is constrained by the landscape. The termination of the algorithm occurs when the change in the water surface between successive iterations is smaller than the specified tolerance.

1.10 Parallelization

The WDPM can be run as a conventional serial program, where one instruction at a time is executed, or in parallel. The parallel processing uses the OpenCL API.

The original WDPM Fortran code employed an algorithm that subdivided the water matrix into mutually exclusive sub-matrices, each sub-matrix being assigned to a separate process. The sub-matrices were separated by two columns to avoid concurrent writes. While this algorithm is every effective with OpenMP, it created problems for OpenCL.

Figure 1.6 illustrates the matrix decomposition used in the OpenMP version of the WDPM code. The colors in the figure represent how the data was split among available processors. In this case, all nodes with the same colour will be processed by the same processor in serial (one after the other) but in parallel (run at the same time) with nodes of other colors in the different region but in the same position of the other sub-matrices.

The Fortran 95 code that processes the nodes is given in Algorithm 1.2. The matrix which holds the water depths is called 'bigwater', as it is 1 array cell larger on each edge than the water matrix, to avoid having to check the boundaries. Similarly, the matrix which holds the DEM is called 'bigdem' as it is also 1 array cell larger on each edge than the original DEM. The code first processes the sub-matrices (referred to as 'slices'). When all slices have been finished, the code then processes the regions in-between the 'slices', which are referred to as 'boundaries'.

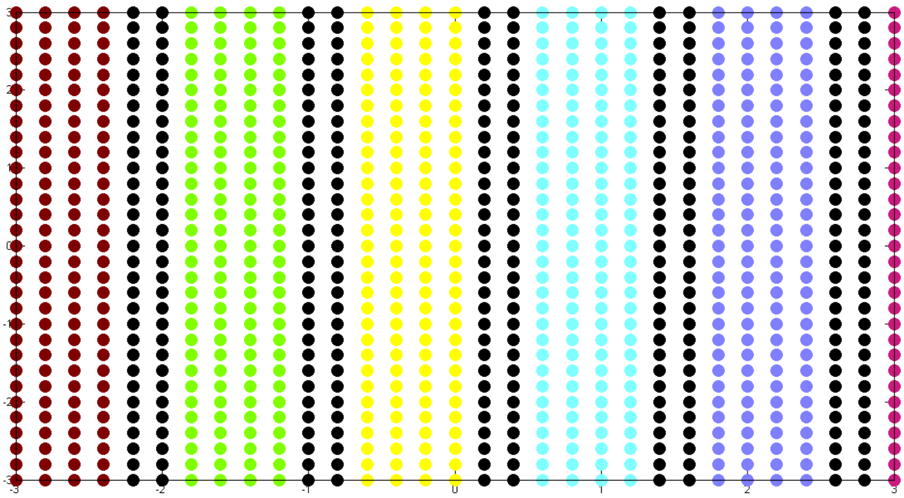

Figure 1.6: Schematic diagram of array decomposition used by OpenMP version of the WDPM.

Algorithm 1.2 Original WDPM Fortran code using OpenMP.

```
! do a set of iterations and then test for convergence
starttime = omp_get_wtime()
do while (.not.(done))
  oldwater = bigwater
  do i = 1,1000
  call omp_set_num_threads(numslices)
  !$OMP PARALLEL
    do row = 2, numrows+1
      do col = startcols(1 + omp_get_thread_num()), endcols(1 +
          omp_get_thread_num())
        if (bigwater(row,col) > 0.0 .and. (bigdem(row,col) >
            missingvalue)) then
          call runoff(row, col)
        end if
      end do ! columns
    end do ! rows
  !$OMP END PARALLEL
  !$OMP BARRIER
  ! do spaces between slices
    if (numslices > 1) then
    call omp_set_num_threads(numboundaries)
    !$OMP PARALLEL
      do row = 2, numrows+1
        do col = boundarystartcols(1 + omp_get_thread_num()),
            boundaryendcols(1 + omp_get_thread_num())
  ! add check for all cells missing
          if (bigwater(row,col) > 0.0 .and. (bigdem(row,col) >
              missingvalue)) then
            call runoff(row, col)
          end if
        end do ! columns
      end do ! rows
    !$OMP END PARALLEL
    end if
  end do ! 1000 iterations
  k = k + 1000
```

OpenCL (Open Computing Language) is the first open, free standard for cross-platform, parallel programming of modern processors found in personal computers, servers and handheld/embedded devices. OpenCL greatly improves speed and responsiveness for a wide spectrum of applications in many categories. The problem caused by the Fortran OpenMP algorithm in OpenCL is that the OpenCL will do more work to get the task done than it is supposed to. The implementation of the Shapiro and Westervelt algorithm was changed by noting that the goal is to process all the nodes without concurrent writing to the same point in the matrix. The entire matrix was divided into 9 slices such that concurrency will not be violated.

Figure 1.7 illustrates the slices used by the OpenCL version of the WDPM, in which every node with the same colour is processed individually by a unique thread. The OpenCl code consists of two pieces, the host and the kernel. The host code sits on the CPU and send instructions to the devices which do the parallel processing. The C code of the Shapiro and Westervelt algorithm host is shown in Algorithm 1.3. The kernel code is executed in parallel by each of the available devices (CPU and/or GPU). The C code of the Shapiro and Westervelt algorithm kernel is shown in 1.4.

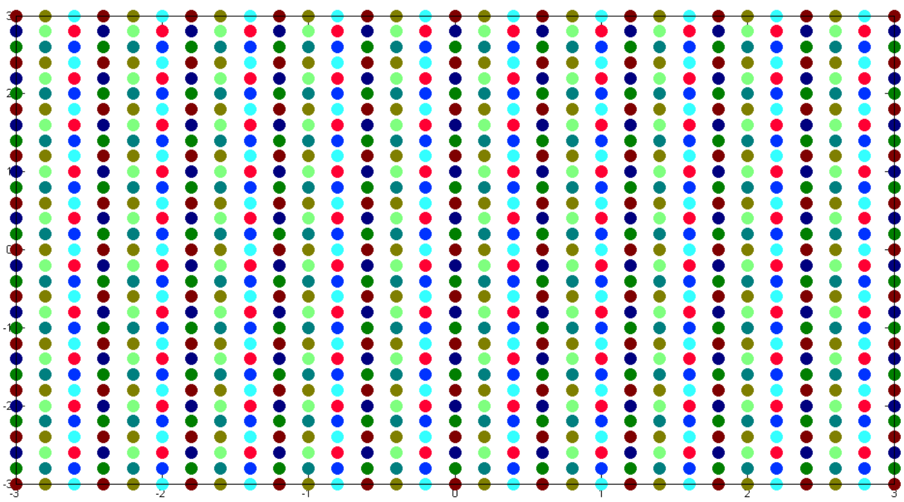

Figure 1.7: Subdivision of the matrix used by the OpenCL version of the WDPM.

Algorithm 1.3 WDPM OpenCL host C code.

```
for (i=0; i<1000; i++){
   for(oi=1; oi<4; oi++){
        for(oj=1; oj<4; oj++){
            err = clSetKernelArg(kernel, 6, sizeof(int), (void *)&oi);

            exitOnFail(err, "set␣kernel␣argument␣oi");
            err = clSetKernelArg(kernel, 7, sizeof(int), (void *)&oj);

            exitOnFail(err, "set␣kernel␣argument␣oj");
            err=clEnqueueNDRangeKernel(queue,kernel,2,NULL,global,NULL
                ,0,NULL,&event);
            exitOnFail(err, "enqueue␣kernel");
            // wait for kernel, this forces execution
            err = clWaitForEvents(1, &event);
            exitOnFail(err, "wait␣for␣enqueue␣kernel");
            clReleaseEvent(event);
          }
     }
     err= clEnqueueReadBuffer(queue,d_bigwater,CL_FALSE,0,bytes,
        bbigwater,0,NULL, &event);
            exitOnFail(err, "read␣bigwater␣from␣device");
            err = clWaitForEvents(1, &event);
            exitOnFail(err, "wait␣for␣read␣bigwater␣from␣device");
            // Reshape Flattened Matrix back to 2D array
            k = k+1000;
}
```

Algorithm 1.4 WDPM OpenCL kernel C code.

```
__kernel void add(__global double *bigwater, __global double *bigdem,
    const double missingvalue, const int numrows, const int numcols,
   const int offset, const int oi, const int oj){
       int row, col;
      int row1 = get_global_id(0);
      int col1 = get_global_id(1);
       int off = offset-1;
      row = (oi-off)+off*row1;
      col = (oj-off)+off*col1;
     if (row>=1 && row<=numrows && col>=1 && col<=numcols &&
       bigwater[row+(numrows+2)*col]>0.0 && bigdem[row+(numrows+2)*
          col]>missingvalue ){
       runoffadd(bigwater,bigdem,row,col,missingvalue,numrows);
    }
}
```

1.11 Zero threshold depth

In the Fortran 95 version of WDPM, shown in 1.2, and in the original C version, the water redistribution is only executed if the depth of water in the centre cell is greater than 0.0. However, this comparison results in many instances of very small depths of water being redistributed. To reduce the run time of the program, a threshold depth has been added to the program as an additional parameter, as described below. Depths of water smaller than the threshold depth are treated as being zero, and are therefore not redistributed. Note that the use of a threshold depth greater than zero will also affect the mean depth of water remaining on the DEM, as calculated by the program, and will also affect the calculated volume of water drained from the DEM. The effects of the zero depth threshold on the execution time and the water distribution

1.12 Potential Issues among DEM datasets for flood hazard mapping

A brief summary of results is provided here for a recent comparison of WDPM runoff maps generated from different digital elevation models (DEMs). The analysis was conducted for two Prairie locations; one located in Saskatchewan and the other in Manitoba. The DEM data sources included products derived from, Light Detection And Ranging (LiDAR), aerial photos, the National Topographic DataBase (NTDB), satellite optical stereo-images, and space-borne radar. The spatial resolution of DEMs obtained for the analysis ranged from 5 m to 30 m.

One site located at Swift Current, SK is characterized by an agricultural region surrounding a valley containing Swift Current Creek; relief = 80 m. The second area located west of Winnipeg, MB is an agricultural area that contains two narrow drainage channels; relief = 8 m. Several gridded DEMs were obtained at both locations for the comparative analysis which included:

- 5 m LiDAR DEMs (Agriculture and Agri-Food Canada)
- 20 m (Manitoba) and 30 m (Saskatchewan) Ortho-DEMs used primarily for the purpose of rectifying orthophotos for a Saskatchewan based digital mapping program (source: SGIC group) and also for similar mapping in Manitoba
- 30 m Canadian Digital Elevation Data, CDED DEMs (source: GeoBase)
- 30 m void filled, downscaled DEMs from the Shuttle Radar Topographic Mission. Specifically SRTM V3 (Source: NASA's Jet Propulsion Laboratory. Currently, these data are no longer available online)
- 30 m ASTER (Advanced Spaceborne Thermal Emission and Reflection Radiometer) Global Digital Elevation Model GDEM and GDEM V2 (Source: NASA's Jet Propulsion Laboratory)

Table 1.2 gives a summary of technical specifications for the DEMs as well as potential issues encountered when using these types of elevation data for flood hazard mapping across entire landscapes. This includes issues related to project resources, systematic errors in elevation data and land surface representation anomalies. Resource issues can be a general limitation simply based on the cost to acquire DEM data and the technical skills required to process massive and complex datasets; e.g. LiDAR point clouds and DEMs. However, systematic data errors and land surface representation issues are more critical concerns for generating realistic runoff maps with WDPM.

Figures 1.8 and 1.9 show examples of WDPM runoff distribution maps generated for the different DEMs. The maps were derived by applying a uniform water depth of 73 mm to all Swift Current DEMs and 116 mm to the Manitoba DEMs. Strictly for reference purposes, the different depths applied are equivalent to the maximum 24 hour accumulated rainfall totals for the 1:100 year return periods at these locations; which also serves to highlight differences in the local climates. Runoff maps are also included for the same water depths applied to the 5 m LiDAR DEMs for relative comparisons against the "expected" runoff distributions.

Contrasts among the runoff maps observed in both cases suggest differences in DEM technical specifications and terrain representation issues are important concerns for flood hazard mapping. The resulting maps highlight potential issues such as terrace-like incoherent runoff patterns generated on CDED surfaces for both landscapes; which are attributed to the use of integer storage formats. Poorly captured terrain features in past versions of Ortho-DEM based products are also a key concern. This type of problem is partly due to the subjectivity of capturing relevant surface features

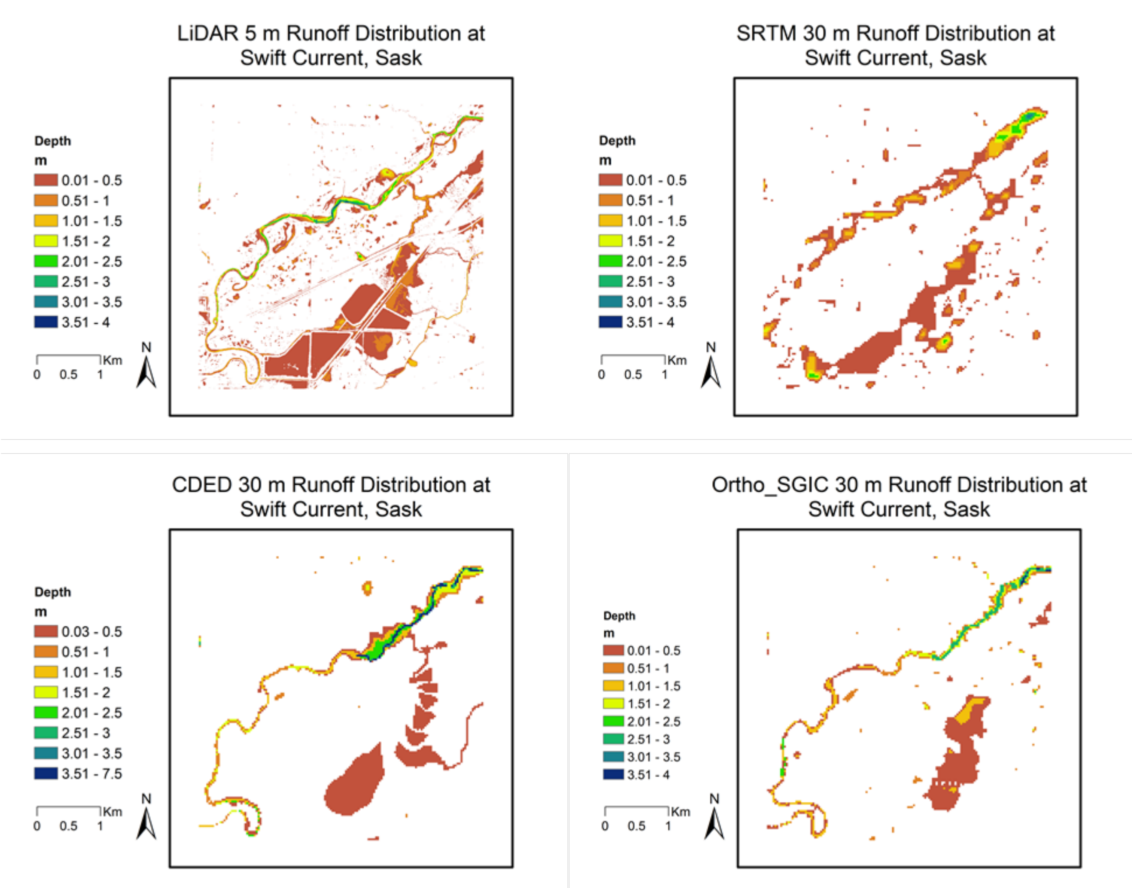

Figure 1.8: Runoff distributions for DEMs near Swift Current SK based on 76 mm of water added.

and systematic issues related to photogrammetric techniques used for developing DEMs, i.e. hardware, software, and data formats.

In more extreme cases, representative land surfaces may even appear nonsensical. Figure 1.9 shows an example based on the SRTM runoff map at the Manitoba location. This critical issue can be attributed to the difficulties of capturing elevation information using space-borne radar or optical techniques over areas of very low relief or dense vegetation canopies, and where limited ground control points exist. This is also a severe problem for ASTER DEMs versions 1 and 2, both of which have been observed to contain large pits in the order of 30 – 50 m deep and "mole-run" artifacts that appear as positive curvilinear surface features (not shown here). Such anomalies can be partly attributed to insufficient scene stacks over a particular region. In such cases, DEMs for these areas have no practical value for overland runoff simulations.

Table 1.2: Technical specifications of DEMs.

DEM Dataset	Cell Resolution (m)	Standard Format	Vertical Accuracy (m)	General Issues
LiDAR	5	real	± 0.3 – 0.5	Cost, data volume and correcting drainage connectivity (e.g. for culvert locations)
Ortho - DEM models	5 – 30	Integer or real	± 1.5	Autocorrelation / interpolation issues; Subjectivity of ancillary data collection; Adequate capture of road networks
CDED	30	Integer	Varies by source data and location	Contour artifacts and integer values result in terraced-like landscapes
SRTM	30 and 90	Real or Integer	± 12 – 16	Winter survey, backscatter (e.g. low relief, dense vegetation) and coarseness
ASTER	30	Integer	± 20	“Mole runs” and 30-50 m deep pits

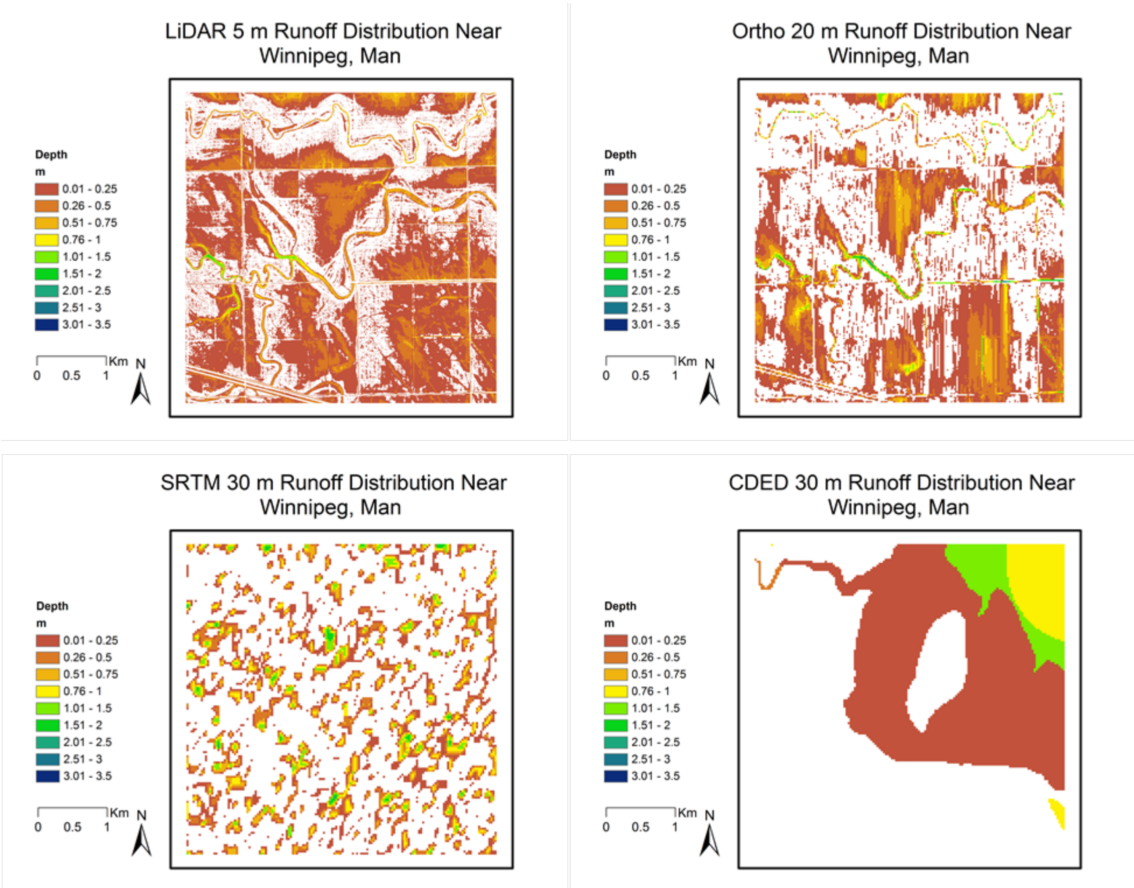

Figure 1.9: Runoff distributions for DEMs near Winnipeg MB based on 116 mm of water added.

2 Installing the WDPM

The source code for the WDPM is available from the GitHub repository at https://github.com/CentreForHydrology/WDPM. Additional documentation and the compiled version of the WDPM for Windows, may be downloaded from the website of the Centre for Hydrology at https://research-groups.usask.ca/hydrology/modelling/wdpm.php.

2.1 Program requirements

Although the program does not require other programs to be installed, it works much better with them. Because these programs change frequently, and to avoid over-writing existing versions with older versions, these programs are not supplied with the WDPM. You must install these programs, preferably before installing the WDPM.

The memory required by the program will depend on the size of the DEM processed. As is described in section 3.4, the primary limitation of the program is its processing requirements.

2.1.1 OpenCL drivers

As described above, the WDPM uses OpenCL to provide parallel processing, which requires the appropriate drivers to be installed. However, the WDPM can also run in serial mode, if you are unable to find appropriate drivers. The selection of the processing mode is shown on on page 24. The drivers installed will depend on your operating system (Linux, MacOS, Windows) as well as your CPU (Intel/AMD) and your GPU (NVIDIA/AMD). Note that OpenCL has been deprecated on MacOS, so there is no guarantee that it will continue to function in the future.

Some of the drivers sources are listed in Table 2.1 below:

Table 2.1: OpenCl CPU and GPU drivers

Type	Manufacturer	Link
CPU	Intel	https://software.seek.intel.com/intel-opencl
	AMD	https://www.amd.com/en/support
GPU	NVIDIA	https://www.nvidia.com/download/index.aspx?lang=en-us
	AMD	https://www.amd.com/en/support

2.1.2 Python 3

The WDPM can be run directly from the command line, but the GUI requires **Python 3** to be installed. On a Linux system, you can install **Python 3** using your package manager. On Windows, you can install **Python 3** from https://www.python.org/downloads/windows/. On MacOS, you can install **Python 3** from https://www.python.org/downloads/mac-osx/ or from https://brew.sh/.

The GUI also requires the library `wxPython` to be installed. You can do this from the **pip** package manager, which can be obtained for Windows or MacOS at https://pypi.org/project/pip/. Linux users can install pip from the package manager, typically by installing the package `python3-pip`.

Pip can then install `wxPython` the using the command

```
pip install -U wxPython
```

or

```
pip3 install -U wxPython
```

depending on your system.

2.1.3 GDAL

As described above, the WDPM GUI can use the program **gdaldem** to create simple maps of water from the WDPM outputs. If you are using Linux, you can install the **GDAL** suite from your package manager. If you are using Windows or MacOS, you can install **GDAL** from https://gdal.org/download.html. Execution of **gdaldem** is done by a shell script, `cmap.bat` (Windows) or `cmap_black.sh` (Linux/MacOS), which must be in the same directory as the WDPM executable and GUI. The script file can be modified if necessary to specify the path to **gdaldem** or any other desired parameters. The file `colormap_black.txt`, which sets the colours used, must also be in the same directory.

2.2 Binary distribution

The WDPM is available as a pre-compiled (binary) version for Windows. The binary distribution of the WDPM was only compiled for Intel processors, and will not work on computers with AMD processors. The steps required to install the program are

1. Install the **OpenCL** drivers **Python 3** and **GDAL** as described above.

2. Copy the file `WDPM2_WIN.zip` to your hard drive and double-click to extract it here. The files will be extracted to a directory called `WDPM`.

3. To run the program, double-click on the file `WDPM.bat`.

2.3 Installing WDPM from the source code

The most recent version of the WDPM source code may be obtained from GitHub at https://github.com/CentreForHydrology/WDPM. The source code is in the folder `/src`.

2.3.1 Installation of the OpenCL libraries

Compiling the source code requires installation of the OpenCL libraries, as described above.

2.3.2 Building WDPM

The WDPM program code can be built using **gcc**, which is a standard part of all Linux distributions, and can be downloaded for Windows or OSX at http://gcc.gnu.org/. If you are building on Windows, you should install **MinGW**, which is a minimalist installation of the GNU tools for Windows, which includes all of the required tools. To install **MinGW**, follow the following steps:

1. You can get **MinGW** from http://www.mingw.org/
2. Download and run the installation program **mingw-get-Setup.exe**.
3. The **gcc** and **g++** compilers will be installed by default. You will also have to select the **make** package as well, by adding all of the `ming32-make` files.
4. After installation of the packages, you must add the path to **MinGW** to the environment path.
5. You can test the installation using the command:

```
gcc -v
```

If the installation is successful, you will see the version information.

2.3.2.1 Building with cmake

The most reliable way to build the WDPM is using the program **cmake**, which is included with Linux (install with your package manager) and **MinGW**. The source code consists of one file, `WDPMCL.c`. The file `CMakeLists.txt` needs to be in the same directory as `WDPMCL.c`, and contains the instructions to produce the file `Makefile`, which is used to build the WDPM.

Building the program takes 2 steps: 1) creating the `Makefile` with **cmake**, and 2) running **make** to build the program.

1. Creating `Makefile`. Assuming that `CMakeLists.txt` is properly configured, the command on Linux or MacOS is simply

```
cmake .
```

On Windows, the command is

```
cmake -G "Unix Makefiles"
```

2. Building the program. Assuming that **cmake** executed correctly, the **make** command is simply

```
make
```

Once **make** has finished, the executable program `WDPMCL` (Linux/MacOS) or `WDPMCL.EXE` (Windows) will be created.

2.3.2.2 Kernel file

The file `runoff.cl` must be present inside the same directory as the compiled WDPM executable. It contains the code which redistributes the water. It is this code which is run in parallel by all cores on the CPU or GPU.

2.3.3 Other files

Use of the GUI requires the file `WDPM-GUI.py` to be installed in the same directory as the compiled executable file, and for Python 3 to be installed. Use of **gdaldem** to create water maps requires the files `cmap.bat` (Windows) or `cmap_black.sh` (Linux/MacOS) in the same directory as the WDPM executable and GUI. The file `colormap_black.txt`, which sets the colours used, must also be in the same directory. The file `test.txt` is a sample parameter text file. The file `basin5.asc` is a sample DEM.

3 How to run the WDPM

The WDPM can be run either from the graphic user interface (GUI) or from the command line. Most users will prefer the GUI, but the command line version is useful to run the WDPM in scripts.

3.1 GUI

The GUI for the WDPM is written in **Python 3**, and can be executed using the command

`python WDPM-GUI.py`

or

`python3 WDPM-GUI.py`

depending on your computer, in the directory containing `WDPM-GUI.py`. If you have installed the Windows binary version of WDPM, then you can execute the program by double-clicking on the file `WDPM.bat`.

The GUI will look slightly different depending on which operating system you are running. The image in Figure 3.1 shows the GUI under Windows. The **File** menu only displays the information about the program, or quits.

Running the program requires the following steps:

1. Click on Browse to set the working directory. The scratch file (if used) and reports.txt will be written to this directory.

2. Select the method: Add, Subtract, Drain or TextFile. Once you have selected the method, the appropriate module components will be activated. If you select TextFile, steps 3 - 9 may be ignored, once the text file has been selected using the Browse button near the bottom of the screen.

3. Set the DEM file name. The DEM must be an ArcGIS ASCII file (.asc).

4. Set the Water (input) file, if it is required. If it is not used, the file name is set to NULL.

5. If a scratch file is defined, the model results will be written to it every 1000 iterations. This is useful if the process might be canceled, as can happen on a shared cluster, or if there is a crash. It also allows you to save the results if you manually terminate the program. However, writing to a scratch file will slow

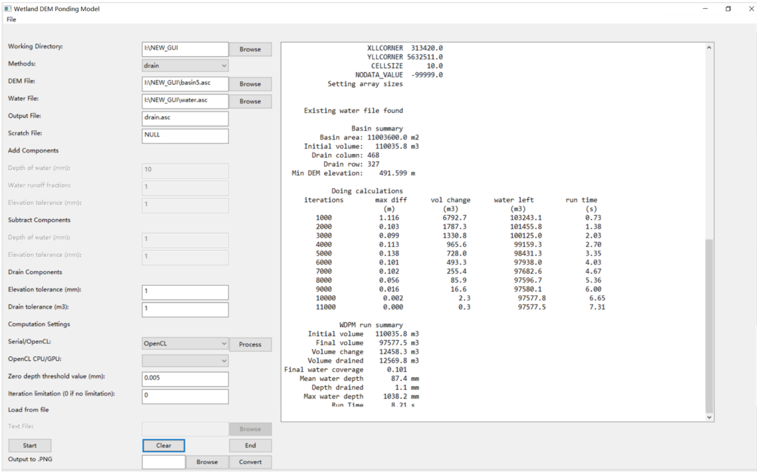

Figure 3.1: The WDPM GUI (Windows version).

the program. The default file name is NULL, which prevents a scratch file from being used.

6. Now set the parameters for the module that you have selected, as described below.

7. Click on Process to set the calculation method and select either Serial or OpenCL processing.

8. If you selected OpenCL, select either CPU or GPU for the parallel processing.

9. Set the Zero depth threshold. This should normally be a very small value like 0.005 mm.

10. Set the Iteration limit. If you don't want to limit the number of iterations, set this value to zero.

11. Click on the Start button to begin execution.

12. When the program has completed execution, you can create a simple image from the output .asc file. Click on the Browse button to choose the file, and then the Convert button.

When the program runs, it outputs to the right-hand side of the screen. The run information includes statistics about the DEM, and the water file (if used). Each 1000 iterations, the model also outputs the change in the water, which allows you to see how quickly it is converging. All of these outputs are stored in the file `report.txt` in the current directory. You can abort the run at any time by clicking on the End button. The Clear button clears the previous run information from the screen.

3.2 Module parameters

In addition to the basic parameters described above, each module has its own set of run parameters.

3.2.1 Add parameters

1. Depth of water (mm). This is the depth of water added to the entire DEM. There is no spatial variability.

2. Water runoff fraction (0-1). Setting the runoff fraction allows some of the water applied to dry ground to infiltrate the soil, while not affecting any of the water applied to existing water (if you are using a water file).

3. Elevation tolerance (mm). If the maximum change in the water depth, at any location in the DEM, is smaller than this value, then the run will terminate.

3.2.2 Subtract parameters

1. Depth of water (mm). This is the depth of water subtracted from the entire DEM. There is no spatial variability, but where the existing water depth is smaller than this value, the resulting depth is set to zero.

2. Elevation tolerance (mm). If the maximum change in the water depth in 1000 iterations, at any location in the DEM, is smaller than this value, then the run will terminate.

3.2.3 Drain parameters

1. Elevation tolerance (mm). If the maximum change in the water depth in 1000 iterations, at any location in the DEM, is smaller than this value, then the run will terminate.

2. Drain tolerance (m^3). If the volume of water draining in 1000 iterations is smaller than this value, then the run will terminate.

Either condition will terminate the run.

3.2.4 Parameter text files

The parameters can also be stored in simple text files. A sample file, `test.txt`, used for the Add module is shown below. The comments must NOT be included. Note that parameters and their order are the same as shown in Section 3.3.

Line	Meaning
`add`	Module to be used
`basin5.asc`	DEM file
`NULL`	Input water file
`water.asc`	Output file
`NULL`	Scratch file
`10.0`	Add 10 mm
`1.0`	Runoff fraction
`1.0`	Elevation tolerance = 1mm
`1`	Use OpenCL
`0`	Use CPU
`0.005`	Zero depth threshold (mm)
`0`	Max number of iterations (0 to omit)

3.3 Command line

Version 2.0 of WDPM can be executed directly from the command line, or from within a scripting language. The command-line arguments are the same for all operating systems. The required arguments are output to the file `report.txt` if the program is run without the correct arguments as shown here:

```
Add module specified
DEM file name (string)
Water file name (string) - Optional, Use NULL to omit Output file name (string)
Scratch file name (string) - Optional, use NULL to omit
Depth of water to add (mm) (real) Water runoff fraction (real)
Elevation tolerance (mm) (real)
Specify 0 for serial CPU and 1 for opencl
Specify 0 for OpenCL CPU and 1 for opencl GPU
Zero depth threshold (mm) (real)
Maximum number of iterations (integer) - Optional, Use 0 to omit

Subtract module specified
DEM file name (string)
Water file name (string) - Optional, Use NULL to omit
Output file name (string)
Scratch file name (string) - Optional, use NULL to omit
Depth of water to remove (mm) (real)
Elevation tolerance (mm) (real)
Specify 0 for serial CPU and 1 for opencl
Specify 0 for OpenCL CPU and 1 for opencl GPU
Zero depth threshold (mm) (real)
Maximum number of iterations (integer) - Optional, Use 0 to omit
```

```
Program arguments in order of specification
Subtract module specified
DEM file name (string)
Water file name (string) - Optional, Use NULL to omit
Output file name (string)
Scratch file name (string) - Optional, use NULL to omit
Depth of water to remove (mm) (real)
Elevation tolerance (mm) (real)
Specify 0 for serial CPU and 1 for opencl
Specify 0 for OpenCL CPU and 1 for opencl GPU
Zero depth threshold (mm) (real)
Maximum number of iterations (integer) - Optional, Use 0 to omit
```

When the program is executed correctly, then the output is passed to the console, and the output is also written to the file `report.txt`.

Example - adding 10 mm of water, without a preexisting water file, or using a scratch file, with a runoff fraction of 1.0, a tolerance of 1 mm, using OpenCl with the CPU, a zero depth threshold of 0.005 mm, and no restriction on the number of iterations on Linux or MacOS:

```
WDPMCL add basin5.asc NULL water.asc NULL 10.0 1.0 1.0 1 0 0.005 0
```

On Windows, the command would be

```
WDPMCL.exe add basin5.asc NULL water.asc NULL 10.0 1.0 1.0 1 0 0.005 0
```

3.4 Model execution time

The WDPM was run on a variety of computers to test its execution time using the same small (471 x 482 elements) DEM, which was used to produce Figures 1.1-1.3. The execution times for all of the modules for all computers are listed in Table 3.1. All of the Add and Subtract runs used a tolerance of 1 mm, with the Drain runs also using a volume tolerance of 10 m^3. In all cases, the zero depth threshold was set to 0.005 mm. No limit was set on the number of iterations. The very poor OpenCL performance of the NVIDIA Quadro K420 GPU is believed to be due to WDPM using double values, which are handled very slowly by this card.

Table 3.2:

Table 3.1: Execution time of the WDPM (seconds).

CPU	GPU	OS	Module	Serial	OpenCL CPU	OpenCL GPU
i7-6700, 3.4 GHz, 8 cores	NVIDIA Quadro K420	Linux Mint 20	Add 10 mm	59.15	18.08	70.81
			Drain	14.69	5.7	19.45
			Subtract 10mm	2.51	1.14	4.15
Xeon W, 3 GHz, 10 cores	AMD Radeon Pro Vega 56	MacOS	Add 10 mm	57.09	33.56	27.9
			Drain	14.73	10.06	10.33
			Subtract 10mm	2.64	1.83	1.85
i7-9750, 2.60 GHz, 6 cores	NVIDIA GeForce GTX 1650	Windows 10	Add 10 mm	57.35	22.65	19.81
			Subtract 10mm	14.80	7.51	7.31
			Subtract 10mm	2.63	1.36	2.00

3.4.1 Effect of zero depth threshold on execution time

The effects of the zero depth threshold were tested by running the program repeatedly on the system with the i7-6700 CPU (3.4 GHz x 8 cores). As can be seen from the execution times listed in Table 3.3, the use of a zero depth threshold greatly reduced the CPU time required to add and drain water, while having small effects on the final volumes of water on the DEM. Using a depth threshold of 1 mm further reduced the execution time, and the number of iterations, but at the cost of reducing the final volume of water on the DEM. It is recommended that a zero depth threshold value of 0.005 mm be used in most cases, particularly for studies where the contributing fraction of a basin is estimated by the volume of water remaining after drainage. If this is not a concern, and the general arrangement of water is the only interest, then a coarser threshold may be justified.

Table 3.3: Execution time of the WDPM for varying zero thresholds.

Module	Zero depth threshold (mm)	Time (s)	Iterations	Final volume (m^3)
Add 50 mm	0.0	5,760.47	385,000	550,1762
	0.005	857.40	385,000	550,1752
	1.0	787.61	378,000	548,3608
Drain	0.0	1,142.87	367,000	380,6052
	0.005	779.85	367,000	380,6051
	1.0	566.82	307,000	380,0840

3.5 Tips and tricks

3.5.1 Estimation of spring flooded areas

As described previously, the WDPM is not a hydrological model. However, it is possible to use the WDPM to estimate the areas of flooding due to spring runoff. The steps are:

1. Obtain an air photo of the region taken in the previous fall or late summer.

2. Establish the initial water distribution. Run the WDPM repeatedly, adding and removing water by trial and error, until the water distribution agrees (more or less) with the air photo.

3. Estimate the total snow accumulation over the winter. Because of sublimation and relocation due to blowing snow, this will invariably be different from the total snowfall. Obviously, this is best estimated by a snow survey. In the absence of a survey, the best method for estimating the snow accumulation is to run the Cold Regions Hydrological Modelling (CRHM) platform, which is available from the Centre for Hydrology. You should be advised that CRHM is not like other hydrological models, and takes some time to learn.

4. The fraction of the snow melt water that infiltrates to the soil is best estimated using CRHM however it can also be estimated using the procedures described in Granger et al. (1984), which can be downloaded from the Centre for Hydrology website at http://www.usask.ca/hydrology/ListPubs.php.

5. Apply the total water equivalent depth of the accumulated snowcover using the WDPM Add module. Set the runoff fraction equal to (1 - the infiltration fraction). The output of the WDPM, (when drained, if necessary) will be an estimate of the spring water distribution.

3.5.2 Contributing fraction

The fraction of the basin contributing flow can be estimated by adding a small depth of water to an existing water file, followed by draining. The contributing fraction is then calculated from the change in volume, which can be obtained from the file `report.txt`. The process is described in Shook and Pomeroy (2011); Shook et al. (2013).

3.5.3 Visualizing the output

The WDPM is intended to be used with a Geographical Information System (GIS) program, as operational use often requires the program output to be overlaid on top of the DEM, and/or infrastructure. GIS programs are also useful for creating the DEM file. There are several Free Open Source Software (FOSS) GIS programs available, such as **QGIS** (www.qgis.org), **SAGA** (http://www.saga-gis.org/en/index.html), and **WhiteBox** (http://www.uoguelph.ca/~hydrogeo/Whitebox/).

The WDPM can also produce a very simple water map from the output file, using the Geospatial Data Abstraction Library (GDAL), which is a free set of command-line GIS tools (GDAL/OGR contributors, 2020). The GDAL program **gdaldem** was used to create the water maps seen in Figures 1.1, 1.2, and 1.3. The use of **gdaldem** requires a color map, which is a text file containing the colors assigned to each depth of water in the output file. The WDPM color map file is `colormap_black.txt`, which must be stored in the same directory as the WDPM executable. The default values in the file are:

```
3,25,0,230
0.001,25,0,230
0,yellow
-9999, black
```

This file will set depths between 0.001 m and 3 m to be blue, zero depths to be yellow, and missing values to be black. The specified depths and/or colors can be changed by simply editing the file.

4 Case Study: Adaptation for the Land and Infrastructure Resiliency Assessment (LIRA)

The Land and Infrastructure Resiliency Assessment (LIRA) project is a sub-component of the recently concluded Climate Adaptation for Resilience in Agriculture (CARA) project funded by Agriculture and Agri-Food Canada (AAFC). The impetus for the LIRA project was based on observed flood damages to vital infrastructure and access to essential services in Honduras in 1998 due to Hurricane Mitch. Although destruction of a similar nature and magnitude is unlikely in most regions of Canada, concerns have increased across the Prairies in recent years due to excess moisture conditions, rising water tables, and severe localized flooding and damage in some locations.

The key goal of LIRA is to assist decision-makers in identifying locations across their landscape that may be vulnerable to extreme overland flood events and potential cost-effective mitigation and adaptation strategies. LIRA is designed to integrate local knowledge, science (e.g. hydrology and climate), GIS technology and economics into a cost-benefit analysis framework. Numerous inputs are required for the economic analysis, just one of which is a flood hazard map. A critical question asked by the LIRA process is whether any economic assets or essential transportation and emergency services routes are located in flood hazard zones; or may be impacted by flood damages incurred elsewhere.

4.1 Flood hazard mapping for LIRA applications

The basic goal of flood hazard mapping is to identify the spatial extents within a landscape that may be covered by flood waters, or potential hot spots where flood damages may be a concern. Flood hazard maps have typically been developed using flood frequency analysis and hydraulic modelling techniques and so have only been possible along primary waterways. Essentially, such products identify flood hazard zones where any developments should be protected, restricted, or not permitted at all. Unfortunately, existing maps such as those developed through the Federal Disaster Reduction Program from the 1980's have fallen into disuse. Effective use of these mapping products would require updates for the recent climate conditions.

More advanced modelling techniques exist today, which can be used to generate flood-risk based hazard maps. Unfortunately, the standard approaches used to develop such maps are incapable of identifying hazards in outlying areas not directly connected to the main drainage channels. The WDPM was introduced to LIRA specifically because of the model's capacity to generate spatially explicit distributed runoff maps across entire landscapes even where no streams exist. A limitation of the current modelling strategy is that the WDPM does not directly model hydrology or hydrodynamics. In the short term, however, application of WDPM has helped to circumvent key difficulties that restrict standard approaches from flood hazard mapping within most Prairie basins.

In recent LIRA pilot studies, the application of WDPM has allowed for diagnostic assessments of runoff accumulation zones and hydrographic connectivity. This has included backwater ponding hazards upstream of road intersections and other potential impoundments which bisect water flow paths. A benefit of this ad hoc approach for LIRA stakeholders has been that diagnostic flood hazard maps for entire landscapes have been provided to decision-makers for the first time. A general distinction between flood hazard zones identified using standard hydraulic methods and those from WDPM is currently defined by the probabilistic analysis of individual flood events and estimates of the resulting water volumes.

4.2 WDPM flood hazard mapping case study results for Redberry Lake Planning Region

Community stakeholder involvement is a crucial resource for LIRA projects. As part of the Redberry Lake pilot study, a LiDAR survey was conducted in the fall of 2011 for the Radisson / Borden region in Saskatchewan to address community concerns over flood vulnerabilities. A validation opportunity for the WDPM flood hazard maps subsequently arose during the spring melt period of 2013 when flood emergencies were declared by both communities. The flooding was attributed to saturated conditions and snow-melt runoff over frozen soils which filled numerous depressions to their storage capacities. Once exceeded, the spilling water activated key flow connections between the depressions, inundating large areas within the communities.

The flooding events were captured by ground and aerial photos taken by federal (AAFC) and provincial agencies (special thanks to Frank Fox; now retired from Water Security Agency). Flood hazard maps generated previously with WDPM, made available to the communities, were compared to the flood event photos. The runoff maps shown here were generated by applying a water depth of 100 mm to the 5 m LiDAR DEMs and are displayed with a GIS overlay of the cadastral fabric for the townships. Figures 4.1 through 4.5 demonstrate the validity of flood hazard mapping using the WDPM based on comparisons of flood extents for large ponded areas and along primary flow paths as seen in the images. These areas are linked by numbered location in the maps and photos.

Figure 4.1 shows results for the Radisson Lake area where water flows along a circuitous course from location 1 to location 4. Figure 4.2 shows results of hazard mapping for the town of Radisson. In this case, an extensive area was flooded just northwest of the town. To partly mitigate flooding, water was pumped from location 1 to 2, and water flowed into the town from the west. The water flowed directly through town along a natural drainage path captured by the LiDAR data (Figure 4.3) and was confined by inflatable barriers which potentially stopped more extensive flooding; as depicted by the hazard map. In Figure 4.3, the water can be seen exiting the town to the southeast from two pathways at locations 1 and 5 and inundating field-sized areas toward the south; a flood path contributing flow toward the southeast was also captured along location 7.

Results for the village of Borden are shown in Figure 4.5. Borden receives excess overland flows from the Radisson area. Water enters Borden from the southwest along location 1 and flows directly through town along location 3 (and to the east toward location 2) to the north where a large ponding area can be seen at location 4. Water then flows southeast through the town along locations 5 – 7 and turns sharply to the east passing under a secondary road (location 8) close to the main highway. Despite the simplicity of the WDPM simulations, the results are surprisingly detailed and the flood hazard map depicts much of the observed flooding in the photo accurately. When a greater depth of water was applied to the DEM, the gap between locations 1 and 2 was filled in more accurately.

The relative accuracy of spatially distributed runoff information for other pilot studies has also been verified against runoff masks classified from remote sensing imagery and, most importantly, the past experiences of community stakeholders. The general accuracy and encouraging feedback from the towns of Radisson and Borden have served to validate the usefulness of a simple diagnostic tool for providing spatially distributed runoff information that has not been previous available. It may be possible to improve upon the methods further by driving the WDPM runoff simulations with estimates generated through a physically-based modelling framework that considers Prairie cold region hydrological process directly.

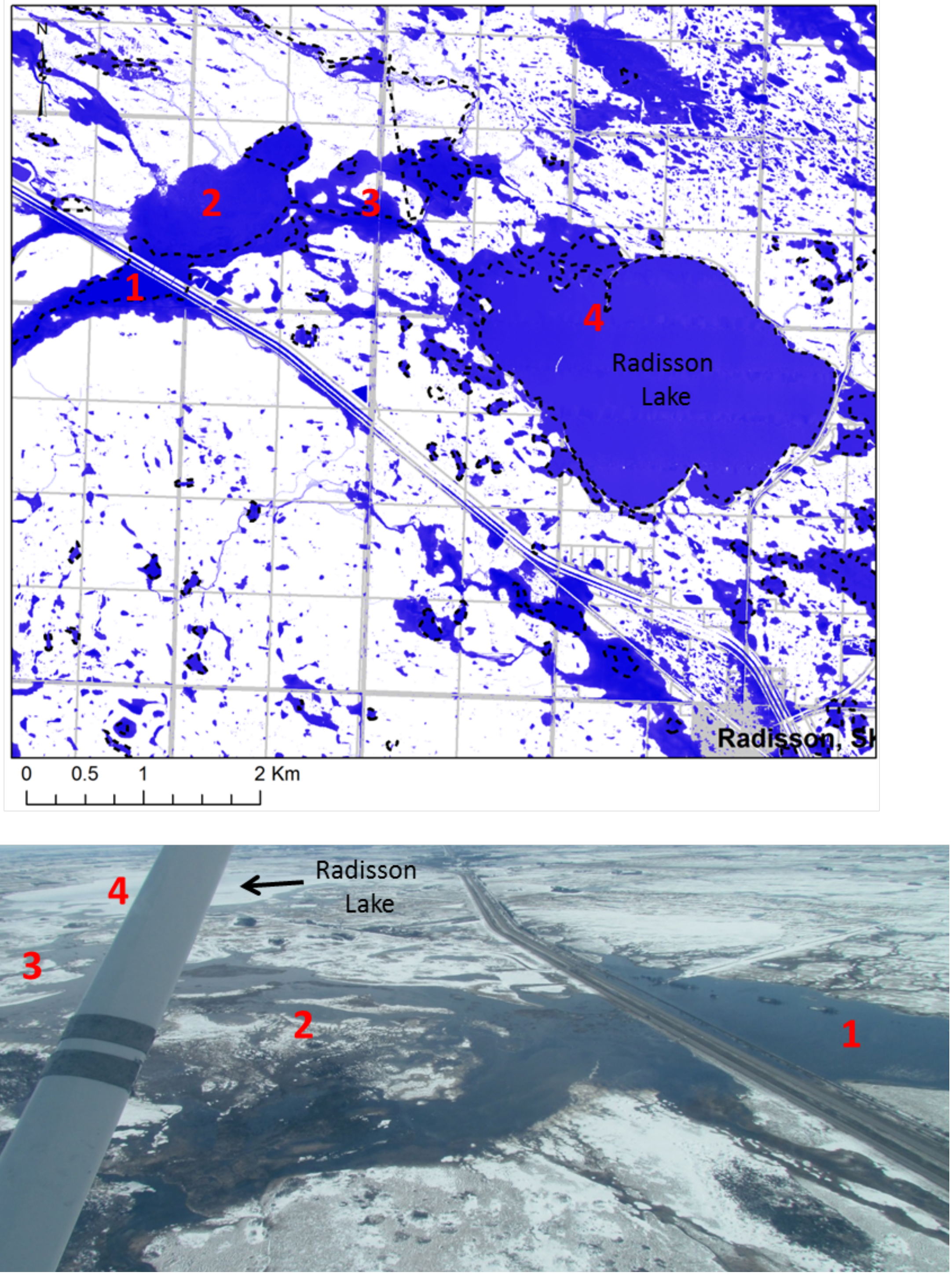

Figure 4.1: Radisson Lake. WDPM output (above) and air photo (below).

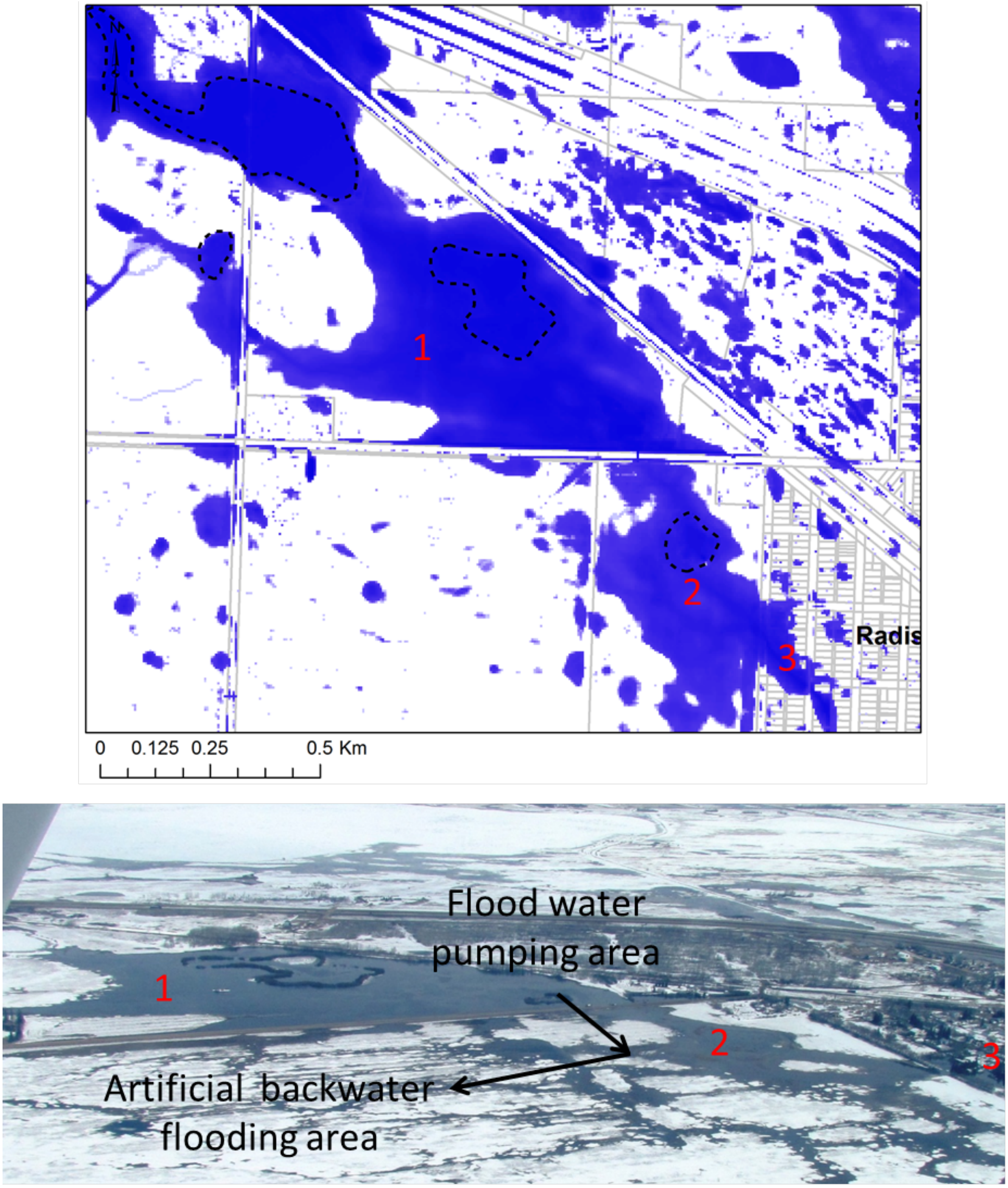

Figure 4.2: Town of Radisson. WDPM output (above) and air photo (below).

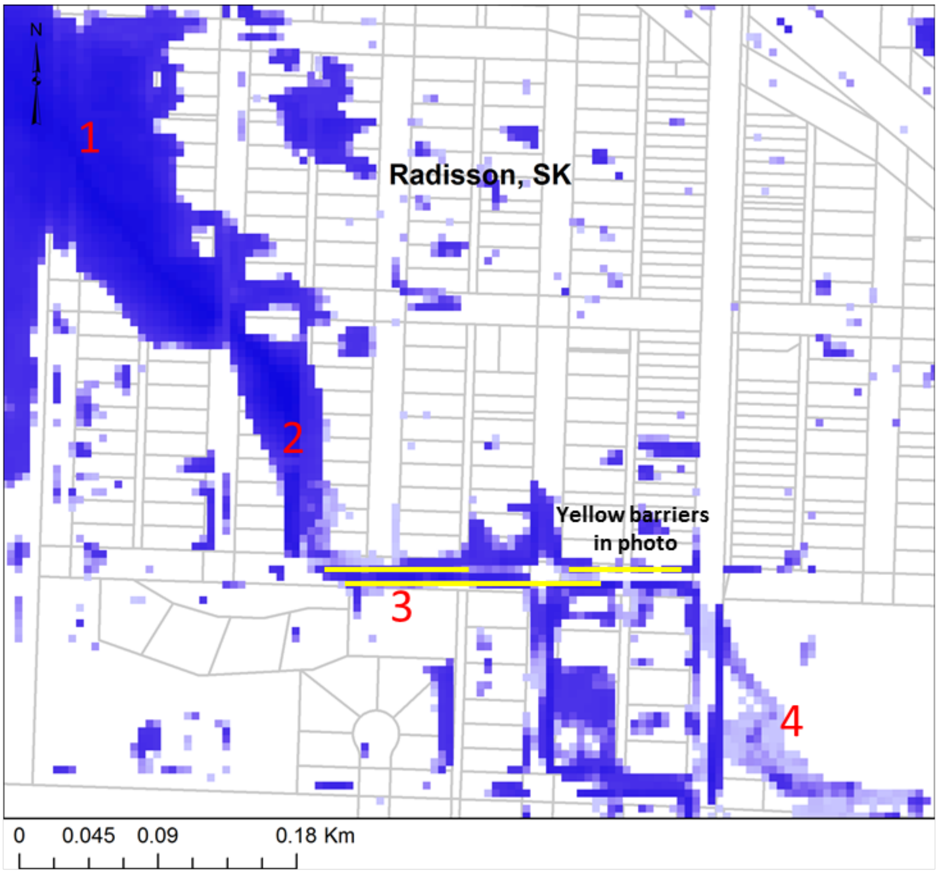

Figure 4.3: Centre of town of Radisson. WDPM output (above) and air photo (below).

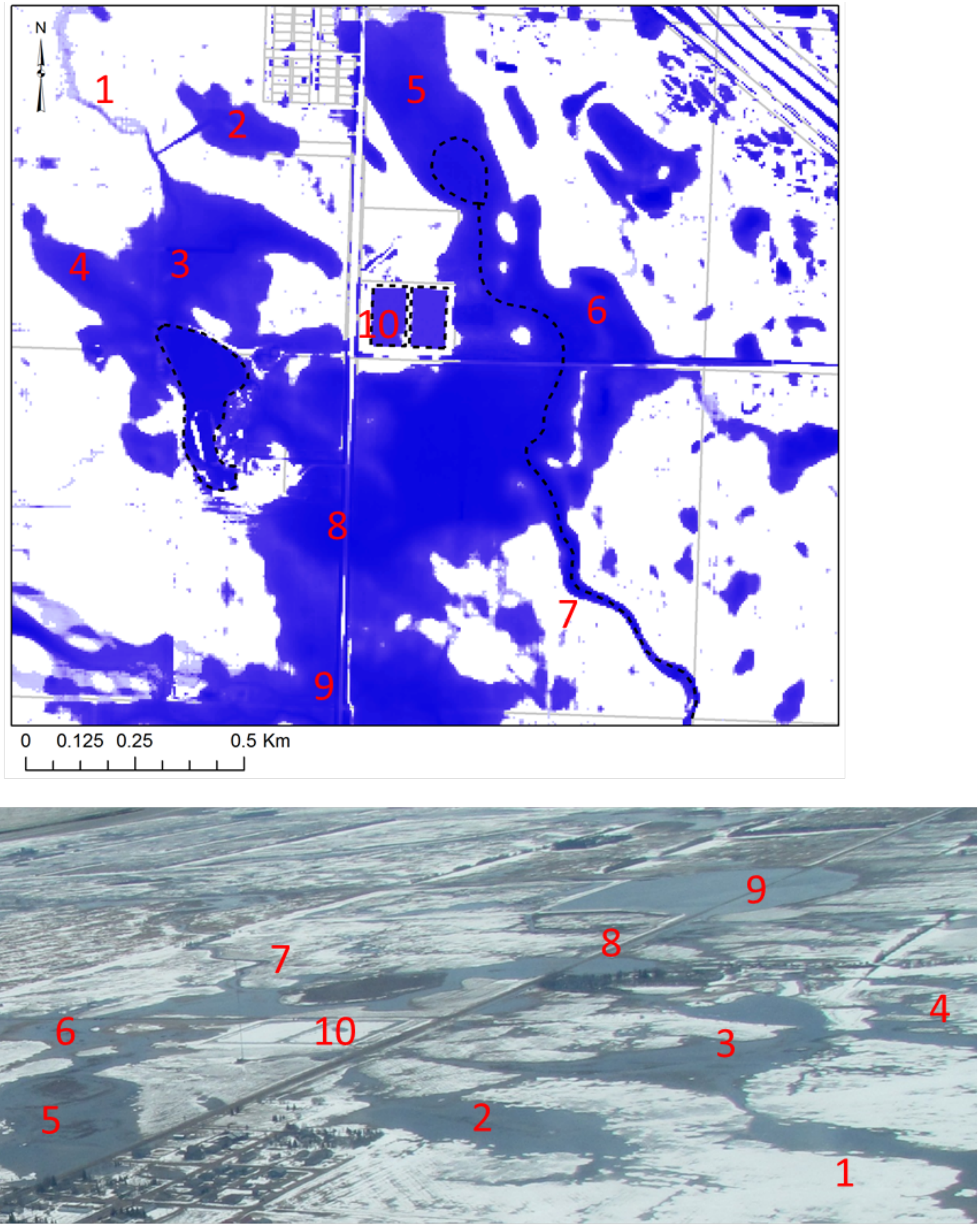

Figure 4.4: South of town of Radisson. WDPM output (above) and air photo (below).

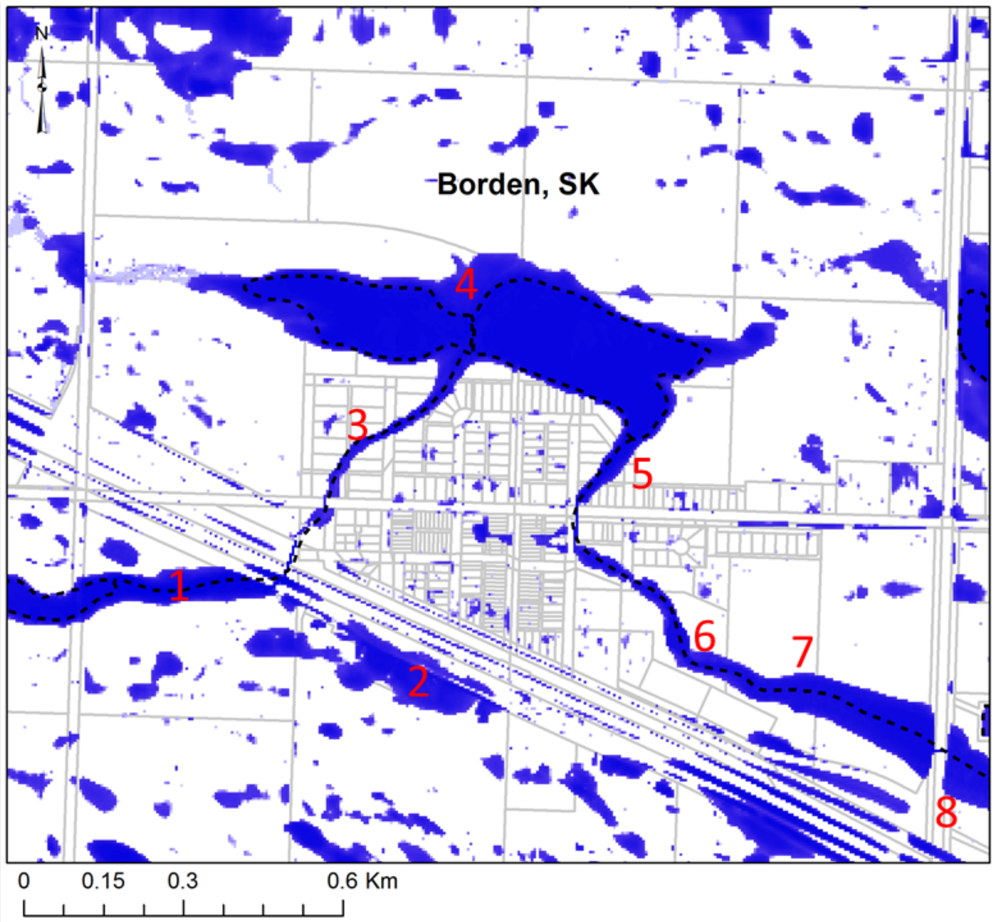

Figure 4.5: Borden region and village.

5 GNU General Public License

GNU GENERAL PUBLIC LICENSE Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections

15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO

LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author> This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

6 GNU Free Documentation License

GNU Free Documentation License Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <http://fsf.org/> Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement. C. State on the Title page the name of the publisher of the Modified Version, as the publisher. D. Preserve all the copyright notices of the Document. E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. H. Include an unaltered copy of this License. I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version. N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section. O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties–for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Bibliography

Costa, D., Shook, K., Spence, C., Elliott, J., Baulch, H., Wilson, H., Pomeroy, J., (2020). Predicting variable contributing areas, hydrological connectivity, and solute transport pathways for a Canadian Prairie basin. Water Resources Research (2020), 2020WR027984. https://doi.org/10.1029/2020WR027984

Garbrecht, J., and L. W. Martz (1997), The assignment of drainage direction over flat surfaces in raster digital elevation models, J. Hydrol., 193(1-4), 204–213, doi:10.1016/S0022-1694(96)03138-1.

GDAL/OGR contributors (2020). GDAL/OGR Geospatial Data Abstraction software Library. Open Source Geospatial Foundation. URL https://gdal.org.

Granger, R. J., D. M. Gray, and G. E. Dyck (1984), Snowmelt Infiltration to frozen Prairie Soils, Can. J. Earth Sci., 21(6), 669–677, doi:10.1016/0148-9062(85)92399-X.

The practice of DEM stream burning revisited. Earth Surface Processes and Landforms 41, 658–668. https://doi.org/10.1002/esp.3888

Pomeroy, J. W., D. M. Gray, T. Brown, N. R. Hedstrom, W. L. Quinton, R. J. Granger, and S. K. Carey (2007), The cold regions hydrological model: a platform for basing process representation and model structure on physical evidence, Hydrol. Process., 21(19), 2650–2667, doi:10.1002/hyp.6787.

Shapiro, M., and J. Westervelt (1992), R.MAPCALC. An Algebra for GIS and Image Processing.

Shook, K. R., and J. W. Pomeroy (2011), Memory effects of depressional storage in Northern Prairie hydrology, Hydrol. Process., 25(25), 3890–3898, doi:10.1002/hyp.8381.

Shook, K., J. W. Pomeroy, C. Spence, and L. Boychuk (2013), Storage dynamics simulations in Prairie wetland hydrology models: evaluation and parameterization, Hydrol. Process., 27(13), 1875–1889, doi:10.1002/hyp.9867.

Index