Digital River Documentation

June 28, 2023

Steven F. Railsback

Contents

Figures

1 Report Objective and Revision History

This report documents the digital river code, which is distributed as a reusable tool for agent-based models in NetLogo.

The digital river is essentially the habitat component of the FYFAM frog model available at: https://ecomodel.humboldt.edu/foothill-yellow-legged-frog-assessment-model and published as supplemental materials to:

Railsback, S. F., B. C. Harvey, S. J. Kupferberg, M. M. Lang, S. McBain, and H. H. J. Welsh. 2016. Modeling potential river management conflicts between frogs and salmonids. Canadian Journal of Fisheries and Aquatic Sciences 73:773-784.

This code represents space as square grid cells. For an alternative using irregular polygon cells (also in NetLogo), see the InSTREAM trout model at: https://ecomodel.humboldt.edu/instream-7-and-insalmo-7.

2 Model Description

2.1 Entities, state variables, and scales

2.1.1 Habitat entities, variables, and scales

River habitat is represented at two scales, reaches and cells. The model represents one "reach", a contiguous section of stream or river and adjacent riparian habitat. A reach is the model's spatial extent, which can be a few 10s of m to 100s of m in stream length. Reaches normally include the full channel width but do not necessarily have to. A reach has a static variable *cell-size* for the width of each of its cells (all cells are assumed to have the same size) and dynamic (time-varying) state variables *step-length*—length of the current time step (in days), *flow*—stream flow (m^3/s), and *temperature*—water temperature (°C). The flow and temperature variables represent averages over the time step.

Cells represent habitat variation within the reach. Cells are square but can provide a fully two-dimensional representation of habitat via techniques such as "warped grids" or simply representing a grid of points on a two-dimensional space. Each cell has an example habitat variable *cell-suitability* to illustrate how such variables can be read in and used.

Cell size (width) is the model's spatial resolution; this spatial resolution *cell-size* is set via a model parameter `patch-width`.

2.1.2 Time scales

The temporal extent of a simulation is set by the user, via the parameters `start-time` and `run-duration`. The parameter `start-time` is a date and time variable; the model starts at the first time in the time-series input file that is at or after `start-time`. The value of `run-duration` is the maximum number of days that the simulation lasts; the model stops at the last time in the file that is before or at `run-duration` days after `start-time`.

The temporal resolution (time step length, reach variable *step-length*) is variable and determined by the flow and temperature input. The model simply executes one time step for each time in the flow and temperature input file (Sect. 3.3.4), and each such time step represents the time until the next time in the file. Flow and temperature values in the file represent conditions from the time associated with them in the input file until the next time step starts. For this example input file, if simulations start on April 1:

```
Time,flow,temperature
4/1/2000 00:00,0.38,5.5
4/2/2000 00:00,0.5,5.8
4/2/2000 07:00,10.0,5.8
4/2/2000 17:00,0.5,5.8
4/3/2000 00:00,0.46,6.4
4/4/2000 00:00,0.83,7
4/5/2000 00:00,0.78,6.2
```

the model's first time step will represent April 1, from midnight to midnight, with a flow of 0.38 m^3/s. On April 2, a mid-day flow pulse is represented; the model's second time step represents midnight to 7:00 a.m. at a flow of 0.5, the third step represents 7:00 to 17:00 at a flow of 10.0, and the fourth represents the rest of the day at 0.5 m^3/s. The remaining time steps are each one full day.

Time steps are typically one day, using daily mean flow and temperature as input. However, sub-daily flow pulses (or reductions) can simply be inserted into the input file. The model can also use input at time steps longer than one day, e.g. weekly mean flows.

Time variables use units of days, unless otherwise noted.

2.2 Process overview and scheduling

The digital river executes only one action.

1. Habitat is updated. The time step's length is determined from input, and reach flow and temperature are updated. The depth and velocity of each cell is calculated from flow, using methods described in Sect. 2.5.1.

2.3 Initialization

Habitat initialization data are provided via input files and include, for each square cell: coordinates of the cell center (in any Euclidian coordinate system), elevation, the values of *cell-suitability*, and lookup tables of depth and velocity as a function of flow (described in Sect. 2.5.1). These data must all be prepared by the user, typically using hydrodynamic models and geographic information systems.

The spatial resolution of the digital river is not necessarily that of the habitat input, which allows habitat input to have different spatial resolutions in different areas. This ability is useful for representing, e.g., broad and uniform floodplains, at a coarser resolution.

The habitat characteristics of each cell are then assigned from the habitat initialization input. That input is provided for points ("nodes") specified in the geometry input file (Sect. 3.3.1).

Each model cell uses the habitat input for the node in the geometry file that is nearest the cell's centroid.

2.4 Input data

There are two time-series inputs: stream flow and water temperature. The input file provides a mean flow and temperature for each time step.

2.5 Submodels

2.5.1 Interpolation of cell depth and velocity

The depth (m) and velocity (m/s) of each cell is updated daily as a function of the reach's flow. This update is conducted via interpolation from lookup tables provided by the user. These tables include a series of flows, spanning the range of simulated flows from low to high, and the depth and velocity of each cell for each flow. The lookup tables are typically generated by hydraulic modeling, though they could be produced directly from extensive field data.

On each simulated day, cell depths and velocities are calculated by interpolating linearly between values in the lookup tables. This interpolation is limited in several ways to deal with lookup table limitations.

First, each cell has a variable *flow-at-wetting* for the lowest flow at which its depth exceeds zero. This flow is identified by extrapolating downwards from the two lowest flows with non-zero depths in the lookup table. This extrapolation is subject to several conditions:

- If depth is non-zero at the lowest flow in the lookup table, *flow-at-wetting* is set to zero.
- If depth is zero at the highest lookup-table flow, *flow-at-wetting* is set to an arbitrary large number.
- If depth is non-zero only at the highest lookup-table flow, *flow-at-wetting* is arbitrarily set to halfway between the two highest flows in the depth lookup table. (In this case it is impossible to interpolate a value of *flow-at-wetting*. Setting *flow-at-wetting* to the highest lookup-table flow causes division by zero during interpolation.)
- If depth decreases instead of increases between the first and second flows with non-zero depths, then *flow-at-wetting* is set to zero but also subject to the next condition.
- If the extrapolated value of *flow-at-wetting* is lower than a lookup-table flow at which depth is zero, *flow-at-wetting* is set to the highest lookup-table flow with zero depth.

At flows equal to or below *flow-at-wetting*, depth and velocity are set to zero. At flows between *flow-at-wetting* and the lowest lookup-table flow with non-zero depth, depths and velocities are interpolated between *flow-at-wetting* and the lowest flow in the tables with non-zero values.

The second limitation is made if the flow is less than the lowest flow in the lookup table (which should be avoided by including very low flows in the table). In this case, depth is extrapolated downwards from the depths at the two lowest flows in the table and set to zero if a negative value is produced; and velocity is interpolated from the velocity at the lowest table flow and zero velocity at zero flow, but set to zero if depth is zero. (In these cases, depth and velocity are still set to zero if flow is at or below the cell's flow at which depth reaches zero.)

Third, if the flow is higher than the highest flow in the lookup tables (also to be avoided by including higher flows in the lookup table) then both depth and velocity are extrapolated upward from their values at the two highest flows in the table. This extrapolation for higher flows does not allow cells dry at the highest flow in the lookup table to become wet at higher flows. It is possible for this extrapolation to produce negative depths or velocities, when a cell has a lower depth or velocity at the highest flow in the table than at the penultimate flow. In this case, negative depths are set to zero but negative velocities cause execution to stop; the problem must be solved by revising the lookup table.

3 Software Guide

3.1 License

The digital river software is copyrighted and licensed under the GNU General Public License (GPL; https://www.gnu.org/licenses/gpl.html), which means it is free software that anyone can use, modify, and re-distribute; but it must remain free. A copy of the license is on the "Info" tab of the model's NetLogo file.

3.2 Installation

Installing the digital river requires two simple steps: installing NetLogo and then copying the model files. The digital river uses the NetLogo modeling platform (Wilensky 1999). NetLogo is free, easy to install, and available for all common operating systems. FYFAM therefore can be used on Windows, Macintosh, and Linux computers. NetLogo is installed by downloading an installer from its web site at Northwestern University: http://ccl.northwestern.edu/netlogo/.

The digital river and these instructions work only with NetLogo versions 6.2 and higher.

3.3 Model files

The digital river uses several input files, plus a file of parameter values. These files must all exist in the same directory as the digital river NetLogo code file. Examples of these files are distributed with the model, but users typically simulate new sites or management scenarios by creating new files.

All the input files are in plain text format; they can be created in spreadsheet software and then saved in either tab-separated or CSV (comma-separated value) format (as specified below for each file type). The files can all be edited with text editors such as Notepad.

All the input files except the parameter file (Sect. 3.3.5) have user-specified names. The names of these files must be provided in the parameter file: after creating or re-naming an input file, the user must edit the parameter file to provide the file name that the model is to use.

3.3.1 Cell geometry file

This file provides the coordinates of the centers of the habitat cells. The geometry must follow these conventions, which are based on UTM coordinates:

- Cell numbers must be positive integers. (0 is not a valid cell number.) However, cell numbers need not be continuous nor in any particular order.

- Coordinates must be in units of meters.
- For display, X coordinates increase from left to right (west to east) and Y coordinates increase from bottom to top (south to north).
- The cells must be a north-south, east-west grid of square cells (or the equivalent in an arbitrary coordinate system): their centers must be evenly spaced in both the X and Y dimensions.
- Cell size is unrestricted: the distance between cell centers can be any number.
- The cells do not need to make up a square space.

The format of the cell geometry file (Figure 1) is: three lines of header information that are ignored by the computer, followed by one line for each cell. These lines contain the cell number and the X and Y coordinates of the cell center. **These values must be separated by spaces or tab characters, not commas.**

```
FYFAM cell geometry input file
Example site. Coordinates are in meters.
CellNum     X      Y
1     210.53      70.31
4     210.53      73.31
2     210.53      71.31
3     210.53      72.31
...
```

Figure 1. Example cell geometry file.

3.3.2 Cell variables file

The cell variables file provides habitat variable values for each cell. Cells are references using the same cell numbers as in the cell geometry file. The format is similar to that of the geometry file: three header lines followed by one line per cell. These data lines contain:

- The cell number,
- Cell elevation (bed elevation, in meters, using any datum; used for display only), and
- A number with the cell's value of *cell-suitability* (Sect. 2.1.1).

The data lines need not be in any particular order. **Values on each line must be separated by spaces or tab characters, not commas.**

```
FYFAM cell variables input file
Example site.
CellNum     Elevation (m)     Cell-suitability
1           103.2       0.0
2           104.3       0.0
3           101.1       0.32
4           100.4       0.41
...
```

Figure 2. Example cell variables file.

3.3.3 Depth and velocity files

The depth and velocity files provide lookup tables of depth/velocity values for a range of flows, for each cell. These files are usually generated from output from a hydraulic model (see Sect. 2.5.1). The number of flows in the tables is not fixed, and the depth and velocity files can each use different flows and different numbers of flows.

The flows must be in units of cubic meters per second, depths in meters, and velocities in meters per second. **The values in the depth and velocity files must be separated by spaces or tabs, not commas.**

The two files have the same format (Figure 3):

- Three header lines that are ignored by the computer;
- One line that contains only a single (integer) number: the number of flows for which depths/velocities are provided;
- One line that contains each of the flows, in ascending order; and
- One line for each cell, containing the cell's depth or velocity for each of the flows.

```
Depth lookup table file, example site
Flows in m3/s, depth in m.
First the number of flows; one row of flows; then depths for each flow at each cell.
12
     0.04   0.06   0.08   0.1    0.15   0.2    0.3    0.4    0.5    0.6    0.8    1
1    0.08   0.1    0.1    0.12   0.13   0.13   0.15   0.16   0.18   0.2    0.21   0.22
2    0      0.02   0.02   0.04   0.05   0.05   0.07   0.08   0.1    0.12   0.13   0.14
3    0      0      0      0.01   0.03   0.04   0.07   0.09   0.15   0.2    0.26   0.35
4    0.01   0.03   0.03   0.05   0.06   0.06   0.08   0.09   0.11   0.13   0.14   0.15
...
```

Figure 3. Example depth file.

3.3.4 Flow and temperature time series file

One file provides the values of the time-series habitat variables that drive the digital river: flow and water temperature. This file also specifies the model's time step and is explained more fully in Sect. 2.1.2. The file must contain values for the entire time period to be simulated; it can also contain times before and after the simulated period, which are ignored.

The time series input file is designed to be maintained in spreadsheet software and (unlike the other input files) uses CSV format. **Values in the data lines of this file must be separated by commas, not spaces or tabs.**

The time series file (Figure 4) can start with as many header lines as desired, each starting with the semicolon character ";". These header lines are ignored by the computer. The next line must contain only the text "Time,flow,temperature". The remaining data lines each contain:

- A date and time, in the "m/d/yyyy h:mm" format (for example, midnight at the start of May 5, 2010 is: 5/5/2010 0:00. One p.m. on the same day is: 5/5/2010 13:00);
- The flow, in cubic meters per second; and
- The water temperature (C°).

```
;Time series input for years 2000-2004
; Contains flow (m3/s), temperature (C)
; DO NOT CHANGE variable names in row 3
; Times must be in m/d/yyyy h:mm format!
Time,flow,temperature
1/1/2000 0:00,0.38,5.5
1/2/2000 0:00,0.5,5.8
1/3/2000 0:00,0.46,6.4
1/4/2000 0:00,0.83,7
...
```

Figure 4. Example flow and temperature time series input file. This example uses daily values, assuming each flow and temperature represents a full day, starting at midnight.

3.3.5 Parameter file

The values of model parameters are set in a file named `parameters.nls`. This file is actually part of the model's NetLogo code: the file contains a NetLogo procedure named `set-parameters` that sets the values of all the model's global variables. The file can be edited by itself, or from within NetLogo (via the "Includes" button on the Code tab).

The parameters file contains many lines with the same format:

```
set parameter-name value ; description.
```

For example:

```
set geom-file-name "BullCreek_Geom.txt"     ; Cell geometry file name
```