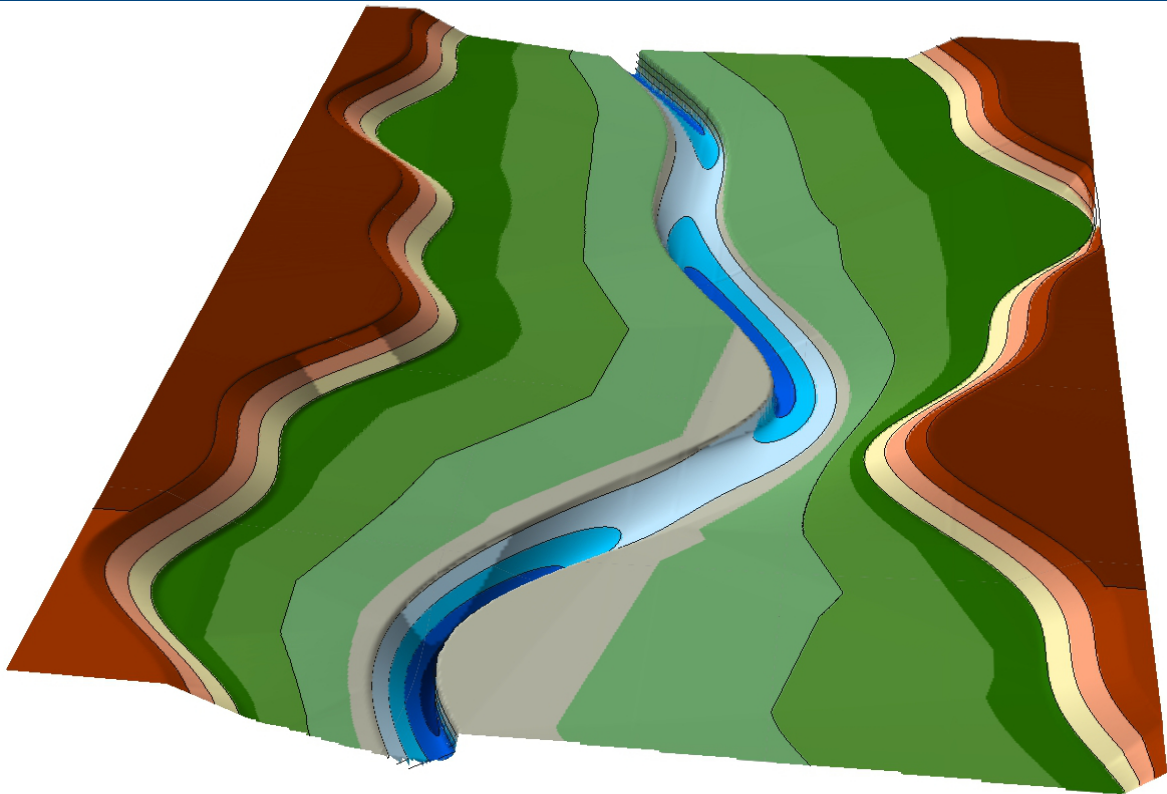


College of Agricultural & Environmental Sciences  
Department of Land, Air, and Water Resources  
UC Davis

# River Builder User's Manual For Version 0.1.1



April  
2018

By: Rafaeli O. Arroyo and  
Gregory B. Pasternack

**UCDAVIS**

**Cite As:** Arroyo, R. O. and Pasternack, G. B. 2018. River Builder User's Manual For Version 0.1.1. University of California, Davis, CA.

Location (URL):

<b>Creators:</b>	Rafaeli Ocampo Arroyo; Gregory Brian Pasternack
<b>Title:</b>	River Builder User's Manual For Version 0.1.1
<b>Publisher:</b>	
<b>Publication year:</b>	2018
<b>Resource type:</b>	Text/Text
<b>Description [Other]:</b>	Software manual for the use of River Builder to create synthetic river valleys.
<b>Subjects:</b>	geomorphology, river topography, river design, river engineering
<b>Rights:</b>	The Regents of the University of California, Davis campus, 2014-17

**DISCLAIMER:** No warranty is expressed or implied regarding the usefulness or completeness of the information contained in this manual or the associated software. References to commercial products do not imply endorsement by the authors. The concepts, materials, and methods presented in this manual are for informational purposes only. The authors have made substantial effort to ensure accuracy, but there is uncertainty and the authors shall not be held liable for calculations and/or decisions made on the basis of application of this software and manual. The information is provided "as is" and anyone who chooses to use the information is responsible for his or her own choices as to what to do with the data.

© The Regents of the University of California, Davis campus, 2014-18.  
For information contact [gpast@ucdavis.edu](mailto:gpast@ucdavis.edu)

## Table of Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>4</b>
1.1	VANILLA RIVERS .....	4
1.2	GEOMORPHIC COVARIANCE STRUCTURES.....	5
1.3	SYNTHETIC RIVER VALLEYS .....	5
1.4	PAST SRV IMPLEMENTATIONS .....	7
1.5	USER'S MANUAL PURPOSE .....	8
<b>2</b>	<b>SOFTWARE DOWNLOAD .....</b>	<b>8</b>
<b>3</b>	<b>INSTALLATION AND SET-UP .....</b>	<b>8</b>
<b>4</b>	<b>RIVER BUILDER WORKFLOW .....</b>	<b>9</b>
<b>5</b>	<b>RUNNING THE PROGRAM .....</b>	<b>10</b>
5.1	RUNNING IN R .....	10
5.2	RUNNING IN RSTUDIO.....	10
<b>6</b>	<b>HOW TO USE RIVER BUILDER .....</b>	<b>11</b>
<b>7</b>	<b>PROGRAM OUTPUTS .....</b>	<b>11</b>
<b>8</b>	<b>PARAMETERS AND INPUTS .....</b>	<b>13</b>
8.1	DOMAIN PARAMETERS (IN METERS).....	13
8.2	DIMENSIONLESS PARAMETERS .....	14
8.3	CHANNEL PARAMETERS (IN METERS).....	14
8.4	FLOODPLAIN PARAMETERS (IN METERS).....	15
8.5	CROSS-SECTIONAL SHAPE .....	16
8.6	SUB-REACH VARIABILITY PARAMETERS .....	16
<b>9</b>	<b>HOW THE TEXT FILE IS PROCESSED .....</b>	<b>17</b>
9.1	DOMAIN, DIMENSIONLESS, CHANNEL, FLOODPLAIN PARAMETERS .....	17
9.2	SUB-REACH VARIABILITY PARAMETERS .....	17
<b>10</b>	<b>USER-DEFINED FUNCTIONS.....</b>	<b>18</b>
10.1	TYPES OF AVAILABLE FUNCTIONS .....	18
10.2	LIST OF AVAILABLE FUNCTIONS.....	18
10.3	DEFINING FUNCTIONS .....	19
10.4	ASSIGNING FUNCTIONS TO SUB-REACH VARIABILITY PARAMETERS.....	19
<b>11</b>	<b>GENERATING 3D RENDERINGS IN ARCGIS® (ARCMAP®) .....</b>	<b>19</b>
<b>12</b>	<b>ACKNOWLEDGEMENTS .....</b>	<b>20</b>
<b>13</b>	<b>REFERENCES .....</b>	<b>21</b>
<b>14</b>	<b>CHANGE LOGS .....</b>	<b>21</b>
14.1	CHANGES FROM 0.1.0 TO 0.1.1 .....	21

# 1 INTRODUCTION

Designing alluvial river channels that behave naturally is a central challenge facing river scientists and engineers as well as animators and video game developers in the 21<sup>st</sup> century. Though organized and responsive to driving forces, rivers exhibit complex patterns and processes from the scale of an individual grain of sediment to that of an entire continent. Despite roughly a century of research it remains highly uncertain as to which patterns and processes are most important to design explicitly versus which ones should be allowed to emerge on their own after construction. Nevertheless, our understanding of the fluvial patterns and processes as well as our ability to quantify them is increasing rapidly. We can now design much more dynamic rivers than ever before.

It is beyond the scope of this manual to explain the entire scientific foundation of this software. This introduction section simply offers a broad overview of the context of river design. The website at <http://pasternack.ucdavis.edu> has a lot of free educational video podcasts about rivers and related topics as well as web pages that provide more explanation of underlying concepts. Much more is available on the internet and in numerous textbooks. Most people will likely seek to just get started with the software and learn on an as-needed basis as they move along, so that is understandable.

## 1.1 *Vanilla Rivers*

Decades of empirical study of longitudinal and lateral transects of alluvial rivers have yielded an explanation for the central tendency of channel geometric variables averaged over a length of 100-1000 channel widths (i.e., the reach scale) to grow or decline with discharge on the basis of mutual adjustment in order to pass the typical water and sediment delivered by the catchment. Such variables include slope, bankfull width, bankfull depth, sinuosity, entrenchment ratio, and median particle size. For any river reach there can be large uncertainties in the average values of geometric variables compared to empirical regional expectations.

A synthetic river designed only according to reach-scale metrics of central tendency is called a “*Vanilla River*” (Figure 1). Although vanilla can be a delicious flavor, it is colloquially considered bland or generic, generally lacking in desirably variations and complexity. This is apropos, because in fact few river processes are driven by the central tendency of reach-scale river metrics. Instead, they are driven by local to reach scale patterns of topographic variability. Thus, few natural rivers look like synthetic Vanilla Rivers and we must turn to a more sophisticated understanding of topographic complexity to design rivers that reflect their natural patterning.

This software is capable of designing Vanilla Rivers, if that is desired. For use in real rivers, we do not recommend stopping there.

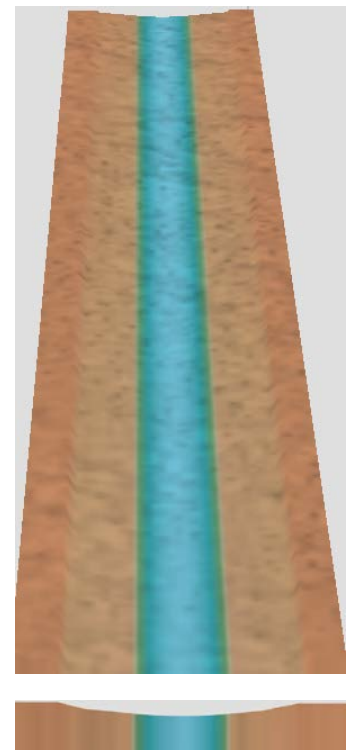


Figure 1. *Vanilla River* example.

## 1.2 *Geomorphic Covariance Structures*

Many measurable variables in geomorphology and allied sciences vary along a pathway, such as a river corridor. Variables could be flow-independent measures of topography, sediment attributes, flow-dependent hydraulics, topographic change, and biotic variables. Lane et al. (2017) reported that for a large region of California river variability metrics distinguished channel types better than traditional central tendency river attributes. These variations can contain some random aspects, but to a large degree they are highly organized, interlocked, and readable (Brown and Pasternack, 2014, 2017). Also, river variations can be layered on top of each other, yielding multiple spatial scales of topographic complexity.

Brown and Pasternack (2014) coined the term "Geomorphic Covariance Structure" (GCS) to mean the bivariate pattern of any two river variables along a pathway. It is not the statistical covariance, which is a single number, but instead a new concept involving the complete bivariate spatial series from which a statistical covariance could be computed if desired. The theory of Geomorphic Covariance Structures (GCSs) is not only useful for assessing the layers of topographic patterning of real rivers (Brown and Pasternack, 2014, 2017) but also for the design of synthetic rivers with more natural landforms that drive the real diversity of physical processes (Brown et al., 2014, 2015).

This software is capable of implementing GCSs to produce rivers with organized, coherent patterns of variability. This is where the real power of this software lies. However, the software does not tell you what GCSs you need to produce different outcomes. You must have in mind what you want and an understanding of what GCS metrics are required to achieve that vision. Over time we will aim to provide an increasing number of template files with different river archetypes you can use as a starting point, but that is still under development. As archetypes become available, they will be posted within the web site at the link below:

<http://pasternack.ucdavis.edu/research/projects/synthetic-river-valleys/>

## 1.3 *Synthetic River Valleys*

Brown et al. (2014) presented a new seven-step method of channel-floodplain design (i.e. synthetic river valley (SRV) design) involving geometric modeling in which multiple scales of continuous equations are specified in each plane (XY, XZ, and YZ) and combined in a digital elevation model (DEM). Figure 2 below combines and simplifies the steps to portray them in a general workflow.

First, you conceptualize the river corridor in terms of its essential elements and scales on the basis of ecogeomorphic goals.

Second, you specify the model domain, including coordinate systems, units, boundaries, and resolution.

Third, you determine the geometric elements for each plane at all scales of interest that are going to be needed, including but not limited to channel centerline longitudinal profile, channel bed elevation longitudinal profile along the centerline, channel width longitudinal profile, XS channel shape (which can be uniform or vary downstream according to a function), longitudinal profile of the outer floodplain boundary on each side of the valley, and the same for optional

terrace boundaries beyond the floodplain on each side. The floodplain on each side of the river is treated independently for greater freedom in design. The first three steps are aggregated into one box in Figure 2.

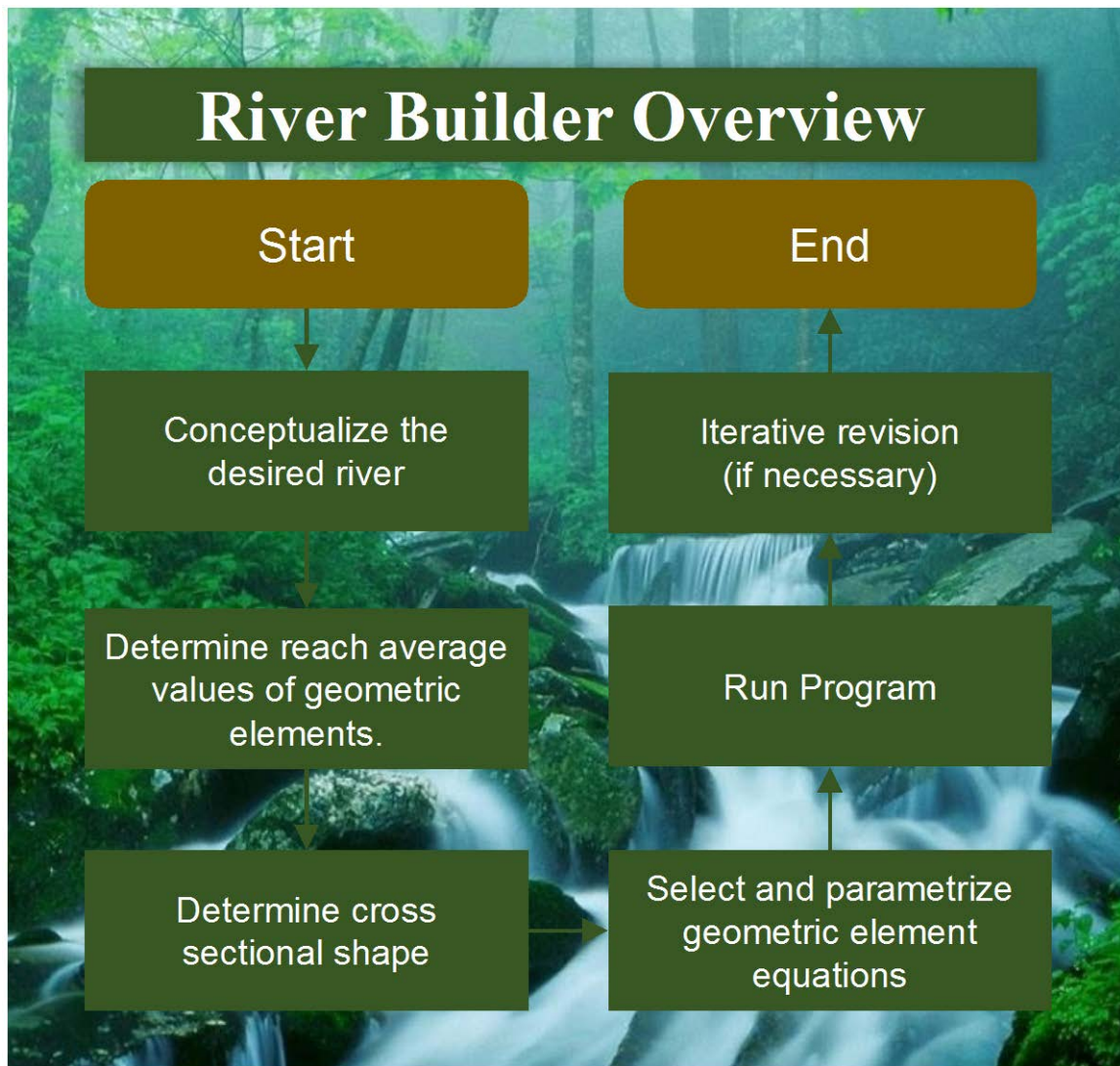


Figure 2. Simplified representation of the workflow for SRV design.

Fourth, you determine reach-average values of geometric elements using standard geomorphic methods. Key values include reach-average bed slope, channel width, floodplain width, median grain size, and either hydraulic radius or bankfull depth. Some variables may be computed from theory and other variables to insure adherence with process concepts, such as computing mean bankfull depth from the Shields equation. Cross-sectional shape is usually often at the reach-scale, but it can be made to vary down a reach. In Figure 2, cross-sectional shape is given its own box, because River Builder software allows you to use a variety of methods for specifying the shape and varying it downstream, so it is not just dependent on reach-average values.

Fifth, you select equations for the longitudinal variations of each geometric element. Although Brown et al. (2014) focused on sinusoids, by now the available functions have expanded to



include lines, curves, sinusoidal oscillations, and Perlin noise profiles. Hopefully in the future there will be cnoidal oscillations, autoregressive meanders, and manually generated polylines and splines (e.g., imported river reach's centerline profile). The decision of which equation to use rests on a firm geomorphic understanding of the local reach, its river, and the region. You can use multiple functions for any one geometric element to obtain complex patterns, especially if the parameters are set to obtain different spatial scaling for each function. For example, one might use a cnoidal function for floodplain sinuosity in a confined valley to represent the river impinging on a bedrock wall and then nest within that a higher frequency sinusoidal function for inset channel sinuosity.

Sixth, you construct the geometric model using some form of software platform. Brown et al. (2014) described how to program an implementation of the theory and provided a Microsoft Excel® version for simplified cases. Past and ongoing software implementations are described in the next section. The River Builder software that this manual describes provides what you need to have a geometric model for many river types and with many features. It does this for you using code programmed in R.

Finally, you parameterize the equations to obtain the locally sized and positioned attributes of the synthetic river. Although parameters for each equation may be specified independently, self-maintenance processes in rivers often produce coherent patterns among multiple geometric elements, including mutual longitudinal variations identified through spectral or wavelet analysis of spatial series or the covariance function between them. Thus, the key decision for parameterization is to decide which two or more variables will be linked to vary coherently. For that to happen the variables have to be described by the same equation, and then parameters governing frequency, amplitude, and phase are specified to yield oscillations that produce the desired landform structure. This is where GCSs come into play. Beyond creating landforms, the deeper goal is to obtain topographically steered, stage-dependent hydraulics that drive a variety of channel maintenance mechanisms when operated on by a natural flow regime. At the same time, there is another goal of providing spatially organized yet heterogeneous aquatic and riparian habitats to serve different species needs in their various life stages.

Once all of this is done, you run the program containing the geometric model to render the synthetic river and get the associated output files. Based on analysis of those outputs, you may then choose to iterate the design to refine and improve it until you have the desired outcome.

## ***1.4 Past SRV Implementations***

The underlying equations needed to make synthetic river valleys are already known, published, and non-proprietary. How those equations are organized into software is what is at issue here. Prior to the development of this R software, SRV algorithms were produced at UC Davis in two different software platforms for different purposes. First, there was the “RiverSynth” approach using Microsoft Excel®. This approach reproduces the examples in Brown et al. (2014) and has received some subsequent development with a few other features.

Second, using an unrestricted donation, my group at UC Davis sponsored World Machine, LLC to incorporate SRVs into the World Machine platform on the hopes that this would make this methodology more accessible. World Machine (<http://www.world-machine.com>) is a native geometric modeling platform for digital terrain development that allows for precise specification

of parameters to design diverse landscapes through dialogue boxes and flowcharting. Implementations of World Machine with river design capabilities may not be available for free and may still be developmental, with little to no technical support. Be sure to touch base with World Machine, LLC to find out the current status of SRV tools in that platform as well as what support they offer to users to help learn their implementation of SRV tools and to address any bugs or problems that come up with your projects.

## **1.5 User's Manual Purpose**

The purpose of this manual is to provide you with an explanation of River Builder software that has the capability to design rivers that meet regional, reach-scale geomorphic expectations, but go far beyond that to have multiple scales and layers of organized sub-reach variability. Certainly, there are still many river archetypes the software cannot yet produce. We welcome future developments that expand the software's functionality more through time.

In most cases we expect people will use this software to design examples of real rivers to yield improved natural outcomes, but in fact the underlying geometric modeling is capable of allowing you to let your imagination run wild and design rivers for other purposes, such as testing dysfunctional river archetypes and creating unnatural, imaginary rivers for fictional purposes (e.g., video games, movies, animations, etc.). Whether you need a digital river design to fix a degraded real river or to save yourself millions of dollars in manual artistic effort for your next 4K resolution open-world fantasy game, this software can be very useful to meet your needs.

## **2 SOFTWARE DOWNLOAD**

River Builder is publicly available for free from the following download locations:

River Builder Website

<http://pasternack.ucdavis.edu/research/model-codes/river-builder/>

This site offers all available resources, including the R package, ArcGIS Toolbox, example river archetypes, and tutorials.

CRAN

<https://cran.r-project.org/package=RiverBuilder>

CRAN offers the essential R package and the optional ArcGIS Toolbox.

We will do our best to release updates to improve the existing software and build in new capacities over time. When we do, they will be available at these two sites.

## **3 INSTALLATION AND SET-UP**

Refer to the following links depending on your operating system to install R and RStudio:

- Linux: <http://www.jason-french.com/blog/2013/03/11/installing-r-in-linux/>
- Windows/OS X: <https://www.andrewheiss.com/blog/2012/04/17/install-r-rstudio-r-commander-windows-osx/>
- 

Note: RStudio is not necessary, but it is highly recommended. It is compatible for Linux, Windows, and OS X and provides an easy-to-use graphical interface for the user to more easily



examine program variables, plots, tables, and access other useful features that are not available on a terminal.

## 4 RIVER BUILDER WORKFLOW

Figure 3 below shows a more complete workflow for this software. It does not show everything in the software, but it gives a much more complete representation of the steps and decision points.

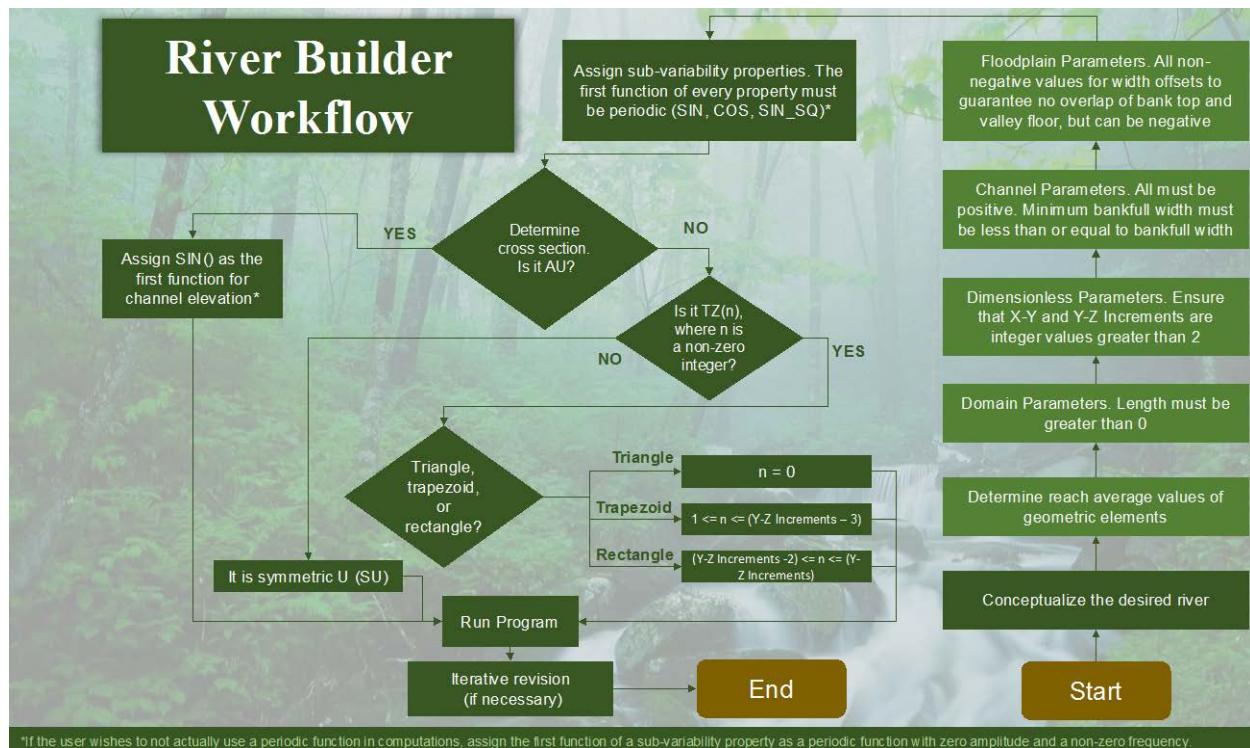


Figure 3. River Builder Workflow.

## 5 RUNNING THE PROGRAM

Have the file “riverbuilder.r” and the input text file in the same directory. There are two ways to run the program:

### 5.1 Running in R

Assuming the most recent version of R is already installed, simply go to the project directory via terminal then execute the following commands:

```
R
source("riverbuilder.r")
riverbuilder(<file name>)
```

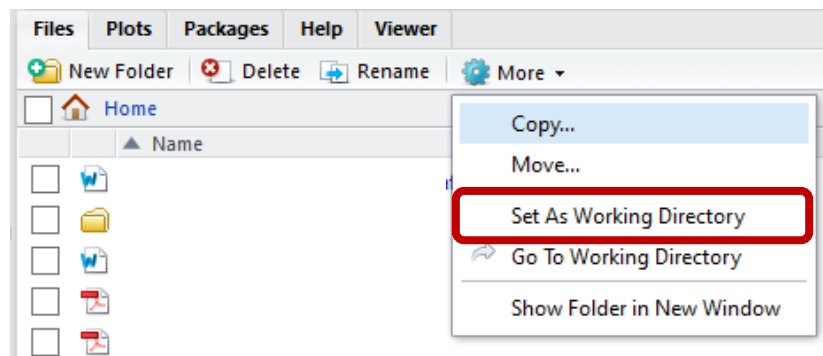
If the last command fails, try this: `riverbuilder("filename.txt", ".", overwrite = TRUE)`

### 5.2 Running in RStudio

In RStudio, first establish the working directory. This can be done by clicking the “...” button on the far right of the interface, as shown in the next page.



A window will pop up in which the user must browse for and select a working directory. Choose the project directory, then click “OK”. Next, click the “More” tab on the interface, then the “Set As Working Directory” option. The project directory is now established as the working directory. Execute the commands ‘source(“riverbuilder.r”)’ then ‘riverbuilder(<file name>)’ on RStudio’s command line, and the program will run.



Note that it may be necessary in RStudio to use a more complete command for River Builder, given as `riverbuilder("<file name>", ".", overwrite = TRUE)`.

## 6 HOW TO USE RIVER BUILDER

The essence of using River Builder comes down to simply entering your design values into the input text file and then running the R script. Its that simple... and yet it is also quite challenging, because you have to know what values you want to put into the input file. The software does not tell you how to specify a river design, but by offering you a set of design options, it certainly focuses your effort on the key components.

To use the software, you have to define reach-average variables with numerical values and establish your sub-variability parameters with functions. The default input text file has pre-defined, acceptable minimum values for the program to function correctly. For full assurance, the user must follow the guidelines specified at the top of the text file. Once the program is run, a prompt appears that tells the user to specify the name of the text file in the current directory to be processed. Several messages will appear on the terminal to indicate the status of the program as well as what CSV and PNG files are generated in the current directory.

## 7 PROGRAM OUTPUTS

### **BoundaryPoints.csv**

Contains point IDs that map to specific points in CartesianCoordinates.csv. These points comprise essential locations along the boundary around the river valley. This is an essential input file used in the ArcGIS Toolbox to make TIN and raster digital elevation models.

### **CartesianCoordinates.csv**

This is the primary output that is the purpose for River Builder. This file contains all of the comma-delimited XYZ coordinates for the synthetic river valley. Points are provided along the key breaklines that define the river corridor as well as along a user-selected number of intervals in the channel. The first row of the file contains the header, {X,Y,Z}. Also, this file is an essential input file used in the ArcGIS River Builder Toolbox to make TIN and raster digital elevation models.

### **Data.csv**

This file contains statistical analysis results from analyzing the river. The information can be organized into four categories.

- Depth and Width statistics: Other data in this file include, in order, the coefficient of variation of bankfull width, the standard deviation of bankfull width, average bankfull width, the coefficient of variation of bankfull depth, the standard deviation of bankfull depth, and average bankfull depth. These values help you understand the effects of the subreach variability functions used for bankfull width and thalweg elevation.
- Caamaño et al. (2009) Criterion: the four parameters in this group are used to assess if any designed riffle-pool couplets will be expected to be self-sustainable on the basis of mean-flow geometry considerations in order to achieve the geomorphic process of flow convergence routing. The four parameters are riffle water surface width (wr), pool water surface width (wp), residual pool depth (hres), and riffle depth to have a velocity reversal (hr). Per Caamaño et al. (2009), "For a given width ratio (wr/wp) and residual pool depth (hres), the water depth over the riffle thalweg (hr) will indicate whether reversal occurs, with deeper flows required for reversal. Furthermore, the riffle must be wider than the pool ( $wr/wp > 1$ ) for velocity reversal to occur." Note that in the quote the symbols for the

variables have been changed from the original to match their occurrence in River Builder. The way this data is used is that the code computes  $w_r$ ,  $w_p$ , and  $h_{res}$  given the subreach variability functions used. Then it computes the minimum thalweg depth value requires. It is then up to the user to evaluate if the actual riffle thalweg depth is greater than this value and if  $w_r/w_p > 1$ . These are not computed for you. Note that given very complex subreach variability functions, these data may not apply to all riffle-pool couplets, in which case manual evaluation of this criterion may be desirable. Finally, note that this criterion is just a rough guide and does not always work; the more non-orthogonal a riffle is, the less accurate it will be. Also, if bed material is different between riffles and pools, then this will not work. For more information, see Jackson et al. (2015).

- Channel Geomorphic Covariance Structure: The values for GCS (-) and GCS (+) help you to understand how bed elevation and width are phased to either other to create either nozzles to oversized channel couplets (-) or wide bar to constricted pool couplets (+).
- Channel Slope & Sinuosity: Most importantly, this file tells you the actual channel slope and channel sinuosity, because for most meandering river scenarios it is extremely difficult to compute *a priori* what the channel slope will be. Recall that the user enters a valley slope in the input file, not the channel slope. Only for a straight channel will the valley and channel slopes be the same. The computed channel slope value can be used to iterate the design as needed to get what you want it to be given a complex subreach meandering function.

#### **CenterlineCurvature.png**

Displays the centerline curvature along the river as defined by the user using the associated subreach variability function, if used. If the river is straight, then curvature is zero. This can be helpful in visualizing how an asymmetric cross-section is going to oscillate back and forth in sync with curvature.

#### **CenterlineCurvature.csv**

A convenient comma-limited text file for using and plotting the centerline curvature data in another program.

#### **ValleySection.png**

Displays the river corridor's cross section at its midway point. The cross section spans the entire valley width. Different colors are used to help you visualize where the slope breaks are.

#### **ValleySection.csv**

A convenient comma-limited text file for using and plotting the valley section data in another program.

#### **GCS.png**

Displays the geometric covariance structures of (i) bankfull width and thalweg elevation; and (ii) thalweg elevation and channel meandering position. The scale on the y-axis is dimensionless, because the geomorphic covariance is the product of two standardized values. When values  $> 0$ , then there is a positive GCS, meaning that variations in the two variables are in-phase. When Values  $< 0$ , then there is a negative GCS, meaning that variations in the two variables are out of phase.

### **GCS.csv**

A convenient comma-limited text file for using and plotting the GCS data in another program.

### **LongitudinalProfile.png**

Displays the elevational series of different variables as they change down the river. Specifically, the profiles of valley top outer edge, valley floor outer edge, bank top, and thalweg elevation.

### **LongitudinalProfile.csv**

A convenient comma-limited text file for using and plotting the longitudinal profile data in another program.

### **Planform.png**

Displays the top-down view of the river to see the width patterns in the river. This plot consists the major breaklines used to define the river's topographic, including the channel centerline, channel banks, floodplain left and right outer edges, and valley top left and right outer edges.

### **Planform.csv**

A convenient comma-limited text file for using and plotting the planform breakline data in another program.

## **8 PARAMETERS AND INPUTS**

### **8.1 Domain Parameters (in meters)**

- **Datum** – a fixed starting point of operation to measure changes of a specific property. Used for thalweg elevation. Input must be a real number. This datum is located at the downstream end of the synthetic river valley, and then the river goes up from there. Thus, if the datum is set as any number  $>0$ , then it is guaranteed that all elevations in the valley will also be  $>0$ . You may set any arbitrary number you wish. If you want to estimate the total elevation range of the thalweg along the river, then simply multiply your Valley Slope input by the Length input and that will give you the overall elevation range of the thalweg. For example, for a 1000 m long channel and a Valley slope of 0.01, then thalweg elevation range is going to span 10 m.
- **Length** – the length of the x-axis of the river valley. Input must be a positive real number. This is only the channel length if the river is straight. If the river is sinuous, then this length is the valley length and the actual channel length will depend on the sinuosity. Sinuosity is computed by the model and reported in the Data.csv file, so you can compute the final channel length if you want to know it.
- **X Resolution** – the resolution of equally spaced data points spaced along the X-axis in units of meters. A value of 1 means 1-m resolution. Input must be a positive number. There is no limit to how fine the point spacing can be, other than computation time for your computer. The smaller this number, the longer the code will take to run. Use your judgement to decide how much detail you need given the scale of your river and the resolution of the sub-reach variability functions you are applying. There is no need for extra detail if the river is easily interpolated with less data.

- **Channel XS Points**– the number of equally spaced points defining the channel's cross section. This is not applied outside the channel, only in it. Beyond the channel, River Builder provides you with points along the major breaklines, and then you can interpolate the terrain between those on your own later using a TIN approach. Input must be a positive integer greater than 1. For field data collection in a U-shaped channel, people commonly use ~ 15-30 points. An odd number is wise if you want to have a line of output points exactly in the center of the channel.

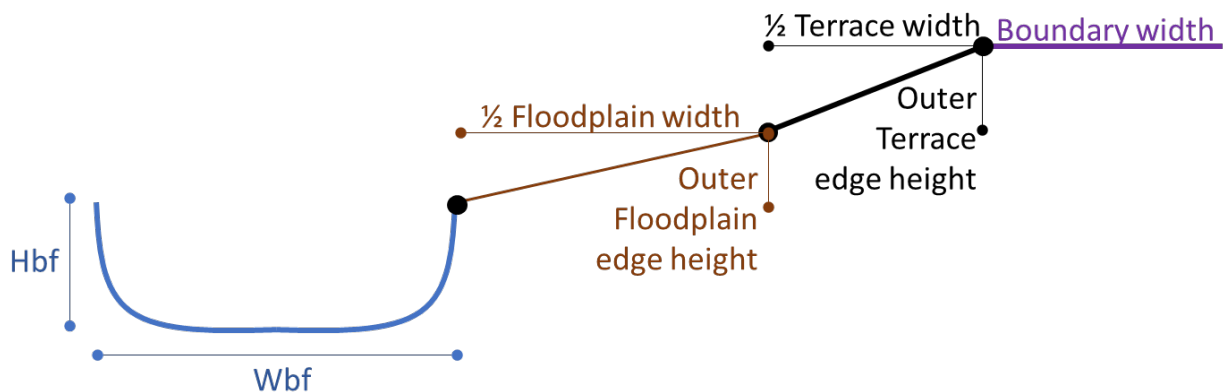
## 8.2 Dimensionless Parameters

- **Valley Slope (Sv)** – the slope of the valley floor along the X-Z plane. It is used to calculate the overall channel slope with a given sinuosity and govern the incline/decline of the surrounding valley. Input must be a real number.
- **Critical Shields stress ( $\tau^*50$ )** – This is an optional input. If you are going to specify a bankfull depth, then set this value to zero. If you want to use the classic equation to establish a bankfull depth to just yield the shear stress needed to mobilize the bed for a specified bed material grain size, then you can use this. Common values used for this parameter are 0.03, 0.045, 0.047, or 0.06, depending on what size fraction you want to insure would be mobilized at bankfull flow. The equation that uses this input is Eq. (4) in Brown et al. (2015). Input must be a positive real number.

## 8.3 Channel Parameters (in meters)

The diagram below illustrates the channel and floodplain parameters described in this section and the next.

### River Builder Cross-Sectional Variable Definitions Schematic



- **Bankfull Width (Wbf)** – the distance between bank tops on the X-Y plane, not counting any sub-reach variability functions. Input must be a positive real number.
- **Bankfull Width Minimum** – the minimum distance between the bank tops on the X-Y plane. Input must be a positive real number. This variable is specified to protect the user from making a mistake with sub-reach variability functions in which the banks would



cross over, yielding an impossible outcome. This insures that the river never has a zero or negative width. If you are confident in what you are doing, then you can set this to zero.

- **Bankfull Depth (Hbf)** – the height between the thalweg elevation and then bank top and thalweg on the X-Z plane, not counting any sub-reach variability functions. Input must be a positive real number. Depending on channel slope and variability functions, the local bankfull depth can differ from this value.
- **Median Sediment Size (D50)** – the particle size of the material comprising the river bed. Input must be a positive real number. This is only used if one has left Hbf=0 and wants to compute the bankfull depth value associated with the initialization of bed material transport for this specified grain size. This value is used in conjunction with the critical Shields stress value as explained earlier.

#### **8.4 Floodplain Parameters (in meters)**

Any floodplain parameter may have the value of zero if you do not wish to use it.

- **Floodplain Width** – double the horizontal distance from the bank top on one side of the river to the outer edge of the floodplain on the same side of the river. Note that if the floodplain has a lateral slope, then this value is still just the horizontal distance, not the slope distance on the floodplain surface. This value is the sum of horizontal floodplain distances for both sides of the river, so keep that in mind when designing your floodplain. Input must be a real number, zero or larger. If this number is zero, then there is no floodplain, which yields a confined valley/canyon scenario. If it is a big number, then there is a very wide floodplain. This applies symmetrically to both sides of the river. If you want asymmetric floodplains, then that is handled with independent Left/Right Floodplain sub-reach variability functions.
- **Outer Floodplain Edge Height** – the incremental increase in elevation from the bank top to the outer edge of the floodplain on the Y-Z plane. Input must be a real number, zero or larger. If this number is close to zero, then the floodplain is essentially flat. If it is high, then the floodplain is sloped.
- **Terrace Width** – double the horizontal distance from the outer edge of the floodplain on one side of the river to the outer edge of the terrace on the same side of the river, making it terrace width. This value is the sum of horizontal terrace distances for both sides of the river, so keep that in mind when designing your terraces. Input must be a real number, zero or larger. This applies symmetrically to both sides of the river.
- **Outer Terrace Edge Height** – the incremental increase in elevation from the outer edge of the floodplain to the outer edge of the terrace on the Y-Z plane. Input must be a real number, zero or larger. If this number is close to zero, then the terrace is essentially flat. If it is high, then the terrace is sloped.
- **Boundary Width** – double the horizontal distance from the outer edge of the terrace on one side of the river to an arbitrary point out on the valley floor on the same side of the river. This allows for a flat, wide domain that could be another terrace or just the open

landscape beyond the alluvial river valley. Input must be a real number, zero or larger.

## 8.5 Cross-Sectional Shape

- The shape of the cross section on the Y-Z plane. Input must be AU, SU, or TZ( $n$ ), where  $n$  is the number of edges that defines the length of the trapezoidal base such that  $0 \leq n \leq$  (Y-Z Increments). *It is recommended that unless you really know what you are doing with AU or are starting from a good AU template file, then only use SU or TZ.* Using AU requires specifying

## 8.6 Sub-Reach Variability Parameters

All inputs for sub-reach variability parameters must be user-defined functions, not constant numbers, mathematical expressions, or anything else.

The available functions that may be used to define sub-reach variability are explained in section 10 below. This section defines the geomorphic aspects of the river that variability functions may be used to control.

- **Meandering Centerline** – governs the shape of the river's tortuous flow path (i.e. meandering or sinuosity) on the X-Y plane. At this time, it is not possible to make gooseneck loops, but the available functions offer a good range of flexibility to mimic diverse sinuosity values. If you don't care about the details of the pattern, but only want to match a sinuosity value, then it is recommended that you use the Perlin function and iteratively adjust the parameters until the sinuosity value in the Data.csv output file matches your target value.
- **Centerline Curvature** – Normally the sub-reach variability functions for centerline curvature would be set equal to those governing the meandering centerline. However, one application of the curvature value is with controlling the asymmetry of a channel's cross-sectional shape. To provide greater flexibility with how a river's XS shape changes along the river, we have created a decoupled centerline curvature variable. This variable has no effect on the channels meandering. It only affects how channel asymmetry varies down the river. For example, let's say that you did not want the deepest part of the cross-section to be at the outer bend of a meander, then you could use this to put the deepest part anywhere you want governed by the available variability functions. That is what this is for. This is a highly complex tool to control compared to others, so only use it if you really know what you are doing with your river design concept.
- **Bankfull Width** – governs the downstream variability in the distance between bank tops on the X-Y plane. If you create an undulating function, be very careful that the banks do not literally cross over to create negative channel widths. We have implemented a minimum bankfull width variable to help avoid this, but it is wise to reason out the amplitude of your width function and not have to rely on that safety net. For longer river segments, you might try a linear function that increases width in the downstream direction, consistent with geomorphic theory. You can find parameters for the downstream width increase in the geomorphic literature quite easily.

- **Thalweg Elevation** – governs undulations in the bed elevation along the thalweg on the X-Z plane. Used to create riffle-pool couplets.
- **Left Floodplain** – whereas the channel is treated as one entity having a width, the floodplains have been segregated into independent units. This function governs variability in the location of the outer edge of the river left (i.e. left looking downstream) floodplain.
- **Right Floodplain** – governs variability in the location of the outer edge of the river left (i.e. left looking downstream) floodplain.
- At present there are no sub-reach variability functions included for terraces.

## 9 How the Text File is Processed

The program parses the text file line by line. In each line, every parameter has a value associated with it by the usage of '='. As stated in the guidelines at the top of the text file, to ensure the program functions, no parameter should be left blank. It is important to note that anything after the symbol '#' on a given line will not be read in by the program, so the user can make use of this to add any desirable notes/comments anywhere on the text file.

### 9.1 Domain, Dimensionless, Channel, Floodplain Parameters

These are read in by the program in a specific order, so the order of these sections and parameters must not be modified in the text file.

### 9.2 Sub-Reach Variability Parameters

These are read in by the program not according to a specific order of parameters, but to the parameter names themselves, which serve as keys to specific river properties. Users can define their own functions each with specified values and assign however many functions they want to any sub-variability parameter. For example, if the user wants to add another function to the meandering centerline, he/she must simply add the line "Meandering Centerline Function=<some function>" below the current list of functions for the meandering centerline. Do not use any mathematical symbols like "+" or "-". Instead, add more lines with individual functions for a given variable and then these will all be added up. The only restrictions for the sub-variability parameters are:

- The first centerline curvature function for just the asymmetric U-shape *must* be a sine function. The reason for this is that the centerline curvature is computed by taking the first and second derivative values of the sine function at a given amplitude, frequency, and phase shift for the asymmetric U-shape only.
- The first function of every sub-variability property must be periodic (sine, cosine, sine<sup>2</sup>) (this applies to centerline curvature as well if the cross-sectional shape is *not* an asymmetric U). This is because some parameters for Data.csv rely on amplitudes and frequencies to be specified. If the user wishes to not actually use a periodic function for a given property while ensuring Data.csv is produced, he/she can simply define the first

periodic function with zero amplitude and a non-zero frequency. The default text file “Input.txt” already does this.

## 10 User-Defined Functions

### 10.1 Types of Available Functions

At present there are 5 different kinds of functions available for you to use to create organized sub-reach variability.

Three types of functions (Sine, Cosine, and Sine<sup>2</sup>) provide periodic oscillations. As simple as those are individually, remember that you can add as many of them as you want into one overall function, using addition, subtraction, etc. As shown in Brown et al. (2014), it is possible to get some very interesting and meaningful variability patterns with 2-4 SIN() functions added together, using different values for frequency, amplitude, and phase. One way to ascertain what those parameter value could be would be to perform harmonic analysis on real series of river variables, such as bed elevation, width, meander centerline, etc. You can extract all the periodic functions you want from that kind of analysis and use those in your design to whatever degree you wish.

Because rivers often do exhibit linear trends in variables, such as width and depth, we have added the linear function for use to create that effect.

Sometimes you want to have a function that is deterministic and oscillating, but non-periodic. The best application for this type of function would be for the meander centerline, because channel alignment is not going to be truly periodic. To achieve this, we provide the Perlin noise function. To find out what sinuosity you get from a particular set of Perlin parameters, you have to run River Builder and look at the Data.csv file. That’s a bit clunky, but it works.

### 10.2 List of Available Functions

- Sine – amplitude, frequency, and phase shift must be defined in the form:
  - o SIN#=SIN(a, f, ps).
- Cosine – amplitude, frequency, and phase shift must be defined in the form:
  - o COS=COS#(a, f, ps).
- Sine<sup>2</sup> – amplitude, frequency, and phase shift must be defined in the form:
  - o SIN\_SQ#=SIN\_SQ(a, f, ps)
- Linear – slope and y-intercept must be defined in the form:
  - o LINE=LINE#(s, y).
- Perlin noise – amplitude and wavelength must be defined in the form:
  - o PERL=PERL#(a, w).

The ‘#’ symbol is a placeholder for a positive integer for assigning a unique set of parameter values to a function variable (ex: SIN1, COS3, LINE2, etc).

In general, amplitude controls the amount displaced from the horizontal axis, while both frequency and wavelength govern the frequency of oscillations present in a wave. Phase shift controls where along the river that an oscillation begins, which allows you to lag one oscillation

relative to others if you wish.

### *10.3 Defining Functions*

The first step is to make a list of all the functions you want to use in your design. You can use any one function for multiple variables if you wish or have unique functions for each variable. You may also have more than one function per variable.

The list goes in the section of the input file names "user-defined functions". A list of functions might be something like this:

- SIN1=SIN(50, 2, 0)
- COS1=COS(100, 1, 0)
- SIN2=SIN(0.25, 1, 1)
- SIN3=SIN(0.3, 0.5, 0)
- COS3=COS(0.2, 2, PI/2)
- SIN4=SIN(2, 2, PI)
- COS4=COS(0.5, 1.25, PI)
- SIN5=SIN(25, 4, PI)
- COS5=COS(75, 1.5, 0)
- SIN6=SIN(55, 1.5, 0)
- COS6=COS(25, 4.5, PI)

### *10.4 Assigning Functions To Sub-Reach Variability Parameters*

The way to assign a function to a sub-reach variability parameter is to simply specify it after the name of the parameter. Suppose SIN5=SIN(20, 4, PI/2). To use this function for, say, the meandering centerline, the user must make the following assignment:

Meandering Centerline Function=SIN5

If you want to have an additive function like SIN4+SIN5, then write them as two separate lines and they will be added together. For example:

Meandering Centerline Function=SIN4  
Meandering Centerline Function=SIN5

Note: Do not modify any of the names of the parameters in the input text files (doing so will result in failure to use the functions or values correctly).

## **11 Generating 3D Renderings in ArcGIS® (ArcMap®)**

After you run River Builder in R or RStudio, then you are going to want to visualize your output. Often this type of design work involves several rounds of iteration. To help with this, we provide an independent ArcGIS Toolbox and associated Python script.

Before attempting to do this workflow, first be sure to look at all the outputs from your model rendering to make sure it worked correctly. Look at all the images and see if they make sense

relative to your expectations. Open the CartesianCoordinates.csv file and scroll through it to make sure all the coordinates are correct. One thing we have seen is that if you run this toolbox and then afterwards you re-run River Builder, then ArcGIS may not refresh properly to look for the new coordinate data properly. In this situation we sometimes we “NA” for some of the coordinates. If you see “NA” for any coordinates, then the ArcToolbox cannot work. Thus, it is best to put each run into its own folder to ensure that ArcGIS thinks of each output file as independent.

Note that the toolbox requires 3D Analyst to work, so if you are not licensed for that model then this is not going to work for you.

Below are the steps necessary to generate a 3D rendering of a river using ArcGIS® and the provided Python script:

1. Open ArcMap®. Look for the “Catalog” tab on the far right of the interface. If it does not show, go to Windows -> Catalog from the top menu to have it opened.
2. In Catalog, locate the directory in which the River Builder toolbox is stored. If the directory is not visible, select “Connect to Folder” from the Catalog toolbar, and select the desired directory.
3. Once the desired directory is visible in Catalog, expand the toolbox it contains, right-click the “River Builder” tool, and go to Properties. Under the Source tab, enter the location of where the Python script “RiverBuilder.py” is stored in your local machine. Click OK.
4. The River Builder tool is now available for use on ArcMap. Select it, and a window will appear. Enter values for the following four inputs:
  - Name: The label applied to each output file, i.e. <Name>\_Layer.shp, <Name>\_Boundary.shp, etc.
  - Output Folder: The location in which an output folder named <Name> will be produced. That folder will contain all the output files generated by the script.
  - XYZ File: The Cartesian coordinates file produced by the R program.
  - Boundary Points: The boundary points file produced by the R program.
5. When all the inputs are filled, click OK, and the script will run. After it is finished running, go to the <Name> folder in the output folder that you specified to access the newly generated shape and raster files.
6. You may now wish to manually use 3D Analyst to create contours of your raster surface. You can set the contour interval to yield ~5-10 contours and that should help with visualizing the surface
7. If you want to see 3D oblique views of your SRV, then you can import the raster DEM and the contours into ArcScene and render a 3D surface per the standard workflow for doing that.

## 12 Acknowledgements

The underlying research used to develop River Builder came from a variety of sources, including UC Davis, the USDA National Institute of Food and Agriculture [Hatch project number #CA-D-LAW-7034-H], the Ticho Foundation (unrestricted donation), US Army Corps of Engineers, and



Yuba County Water Agency. Dr. Rocko Brown and Dr. Greg Pasternack also contributed their own resources in developing different aspects of the underlying research. Rafaeli Arroyo programmed River Builder versions 0.1.0 and 0.1.1.

## 13 References

- Brown, R. A., Pasternack, G. B. 2014. Hydrologic and topographic variability modulate channel change in mountain rivers. *Journal of Hydrology* 510: 551-564.
- Brown, R. A., Pasternack, G. B. 2017. Bed and width oscillations form coherent patterns in a partially confined, regulated gravel–cobble-bedded river adjusting to anthropogenic disturbances, *Earth Surface Dynamics*, 5, 1-20, doi:10.5194/esurf-5-1-2017.
- Brown, R. A., Pasternack, G. B., Lin, T. 2015. The topographic design of river channels for form-process linkages for river restoration. *Environmental Management*, 57 (4): 929-942. doi: 10.1007/s00267-015-0648-0
- Brown, R. A., Pasternack, G. B., Wallender, W. W. 2014. Synthetic river valleys: creating prescribed topography for form-process inquiry and river rehabilitation design. *Geomorphology* 214: 40-55. 10.1016/j.geomorph.2014.02.025.
- Caamaño, D., Goodwin, P., Buffington, J. M.; Liou, J. C. P.; Daley-Laursen, S. 2009. Unifying criterion for the velocity reversal hypothesis in gravel-bed rivers. *Journal of Hydraulic Engineering*. 135(1): 66-70.
- Jackson, J. R., Pasternack, G. B., Wheaton, J. M. 2015. Virtual manipulation of topography to test potential pool-riffle maintenance mechanisms. *Geomorphology* 228: 617-627. <http://dx.doi.org/10.1016/j.geomorph.2014.10.016>.
- Lane, B.A., Pasternack, G.B., Dahlke, H.E., Sandoval-Solis, S. 2017. The role of topographic variability in river channel classification. *Physical Progress in Geography*. 18 doi:10.1177/0309133317718133.

## 14 Change Logs

With each update to River Builder there is a list of changes that have been implemented to improve it.

### 14.1 Changes from 0.1.0 to 0.1.1

Most of the changes in this update involve simple improvements to aid with understanding River builder, but there are some substantive additional features. Items in **brown** below are notable new features.

- Removed "\_" from parameter variables in Input.txt
- Added "Bankfull Width Minimum" parameter to Input.txt
- Corrected "Centerline Alignment" as "Centerline Curvature"

- Renamed "Bottom Width Offset" as "Floodplain Width"
- Renamed "Bottom Height" as "Outer Floodplain Edge Height"
- Renamed "Top Width Offset" as "Terrace Width"
- Renamed "Top Height" as "Outer Terrace Edge Height"
- Renamed Y-axis on GCS.png from "Correlation" to "Geomorphic Covariance"
- Allowed bankfull depth to be either (A) user-defined or (B) calculated from critical Shields stress and median sediment size
- Renamed "X-Y Increments" to "X Resolution" and changed the definition of what it is from number of increments to the resolution desired.
- Renamed "Y-Z Increments" to "Channel XS Points"
- Allowed designation of X Resolution along the channel length
- Renamed the output file "CrossSection.png" to be "ValleySection.png" and implemented the section spanning the whole valley instead of just the channel
- Added floodplain, terrace, and boundary visualizations to ValleySection.png
- Automated generation of CSV files corresponding to each output graph
- Updated user's manual.