## Sprint 7 –hydraulic erosion 06/04/2019 – 01/05/2019

### Abstract

The goal of this sprint is to implement the hydraulic erosion algorithm to simulate real water erosion on the terrain mesh. Not only this is one of the largest sprints but also one of the most important ones. Optimisation techniques are discussed within this sprint.

The end goal of this sprint is the ability for the user control the quantity of erosion created by the water particles and a terrain mesh with visible erosion effects.

### Research

#### What is erosion?

##### Introduction

Erosion is a process that occurs in nature. This process is responsible for removing sediment from certain materials, like soil or rock, and carry this sediment to another location.

Erosion can be divided into different types: water erosion, glacial erosion, snow erosion, wind erosion, zoogenic erosion and anthropogenic erosion.

Due to the fact that the erosion algorithm for this project is going to be water erosion, all the other erosion types are not going to be covered in this document.

##### Water Erosion

###### Introduction

Water erosion happens when the particles of water come in contact with a saturated soil or in a greater rate at which water can infiltrate. It is composed by two process, the first is the erosion of the soil and the second is transporting the soil into the flow and start the deposition process.

Water erosion can be divided into four different types: sheet erosion, rill erosion, gully erosion and streambank erosion.

###### Sheet Erosion

Sheet erosion occurs in soils where the surface is mainly smooth and with a uniform slope. This type of erosion is responsible for removing uniform thin layers of soil on the entire field during a long period of time. So, it is possible to imagine that after a few years of removing thin layers, the terrain can be eroded by a significant amount (Fairbridge 1968).

Two basic erosion processes are involved in this type of erosion. First, the water collects the particles of soil and stores them as sediment into the water flow and second the particles are then transported away from the its original location to be deposited afterwards as sediment.

###### Rill erosion

Rill erosion occurs when the water from rain accumulates on a top of a terrain and then begins to flow, using the path of least resistance, creating paths or rills on the terrain itself [Figure 1].



Figure 1 - Milford. Rill Erosion [digital image] [viewed 08 April 2019]. Available from: https://milford.nserl.purdue.edu/weppdocs/overview/images/rillb.gif

The amount of sediment removed from the soil is proportional to the square velocity of the water flow. So, in other words, this means that if the velocity is 4 feet per second then the capacity of removing sediment from the soil is going to be sixteen times greater. To conclude is possible to see that the greater the water velocity the greater the sediment capacity from the water is going to be (Food and Agriculture Organization 1978).

###### Gully Erosion

This type of erosion occurs when the water flow erodes the soil in a way that creates large ditches [Figure 2]. Gully erosion is dependant of large quantities of water for supplying the necessary energy for removing and transporting the sediments.



Figure 2 - Киля, К., 2008. Gully in the Kharkov region in a small cut forest [digital image] [viewed 22 March 2019]. Available from:

https://en.wikipedia.org/wiki/Gully#/media/File:Gully\_in\_the\_Kharkov\_region.jpg

###### Streambank Erosion

Streambank erosion is a process that occurs in nature when the water flow can exert a stronger force than the soil around. Normally if the water flow passes through a vegetated bank the quantity of erosion is going to be lower, this lies on the fact that the vegetation creates a resistance force against the water flow reducing the erosion. However, if the bank around the water flow does not have vegetation to create a resistance force, the erosion can be brutal (Klausmeyer 2015; Reckendorf 2010).

The soil around the water flow is known as banks of the stream (Leopold, Wolman and Miller 1995).

In the [Figure 3] is possible to see an example of streambank erosion.



Figure 3 - McQueeney, C., 2017. Eroding streambank causes property loss and degraded habitat for fish and wildlife [digital image] [viewed 10 April 2019]. Available from: https://i0.wp.com/conservationdistrict.org/wp-content/uploads/Bank\_erosion\_Milk\_

###### Instant Terra

Instant Terra is a 3D application made in C++ for Procedural Terrain Generation. This application was used to get an idea in what should be expected from hydraulic erosion [Figure 4] [Figure 5] [Figure 6].

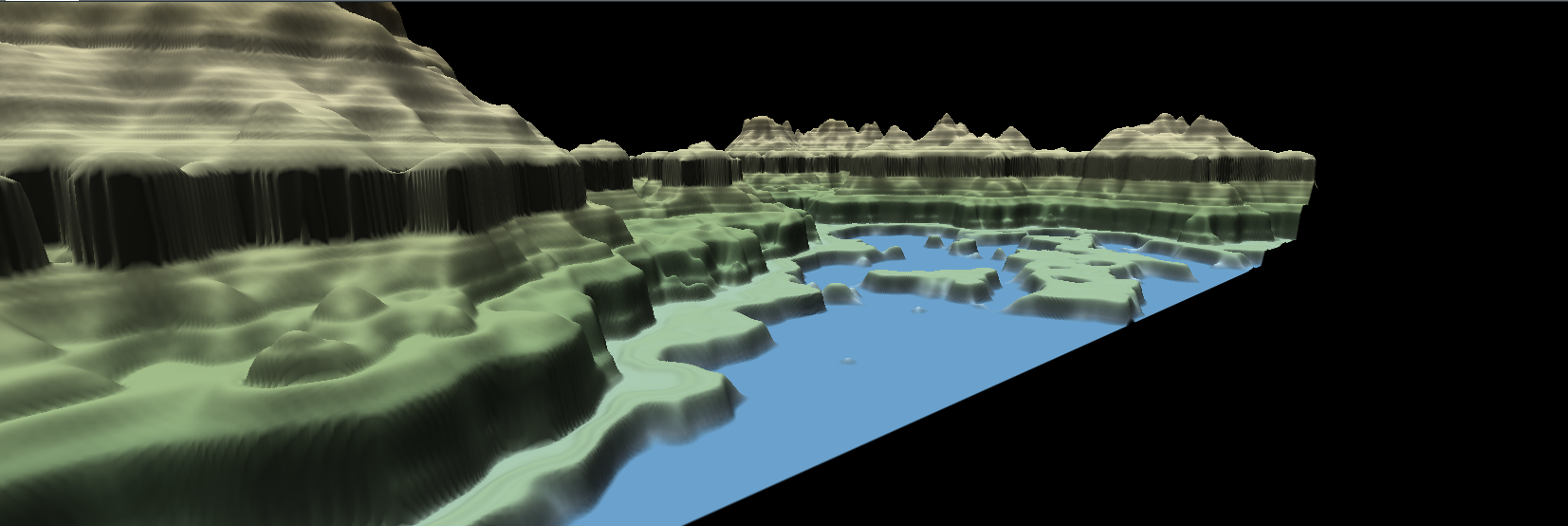


Figure 4 - Instant Terra, terrain without hydraulic erosion

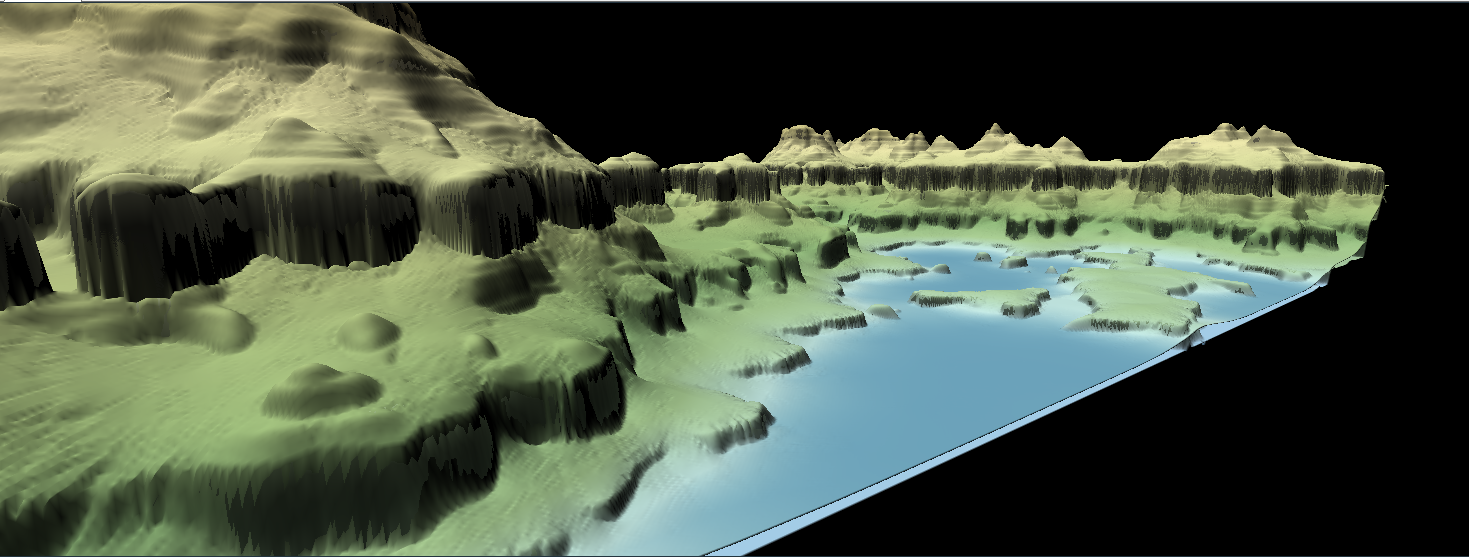


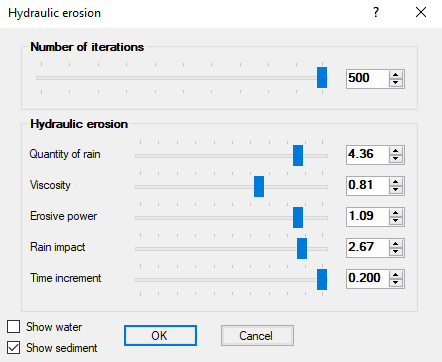
Figure 5 - Instant Terra, terrain with hydraulic erosion

Figure 6 - Instant Terra, hydraulic erosion settings

The algorithm that the author of this project created is inspired in the water erosion algorithm create by (Volynskov, 2011). It is based on the particle simulation in which every single water particle is placed into the map, where every particle as a certain radius with the capacity to erode the terrain around it and afterwards bringing the sediment to the path of least resistance.

### Implementation

#### Algorithm

##### Water Particles

Each particle uses the gradient function to get a direction. If the gradient is zero, a random direction is generated.

The drops are simulated in 2.5D which means the drop is always considered to be at ground level, and only saves its position in two dimensions.

* Particles do not interact
* Do not use the physical principles
* Particles move the same distance every simulation step, although they are not bound to the grid.
* The simulated time per step is not consistent.
* Not suitable to simulate fluids visually.

###### Understanding Water Erosion

Water erosion is composed by two different steps, the first step is erosion and the second is deposition.

In the first step the flow of water starts its erosive activity when going through the land in a steep gradient. When the gradient starts to get a value near zero or the water flow reached the limited amount of soil that can carry then the water starts the deposition process. Normally it is possible to see the most amount of erosion when the flow of water it is at highest speed and the amount of water have a low quantity of sediment.

This amount of soil depends on the following variables:

* Surface Slope
* Amount of water
* Speed of the flow

Untitled picture.png 
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings


Figure - Water Flow

Erosion

* + If the quantity of soil is less than the maximum quantity of soil, then erosion happens.
  + This process removes soil from the terrain and adds it to the flow.

Deposition

* + If the water flow carries more than the maximum quantity then deposition happens, dropping extra carried soil as sediment.

#### Erosion Class

When sketching the erosion class, the author wanted to create a particle inside the erosion method [Figure 8]. Each particle is created inside two for loops one that checks for the number of iterations given by the user and the other for the lifetime of each particle. Each particle consists on the following:

* Vector2 Position
* Vector2 direction
* Float Water Velocity
* Float Water amount
* Float Sediment



Figure - Erode function Sketch

##### Erosion Method

###### 1)Particle Creation

The first step was to create a particle of water at a random point in the map [Figure 9], to be able to do this the author used the Random class from C# (Microsoft .Net 2018).

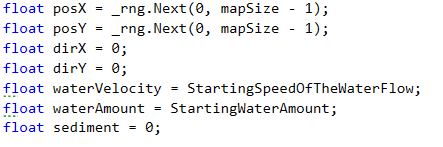


Figure - Particle Creation

###### 2)Converting the Particle Position and creating an Index for the particle

When the position of the particle is defined, a for loop for the lifetime of the particle is created. In the beginning of this for loop the integral position of the particle is saved inside the node variable [Figure 10].



Figure - Passing the variable position to an integral value named node position

Giving some random numbers like the ones defined in the image above, is possible to know exactly were the particle is located at [Figure 11].

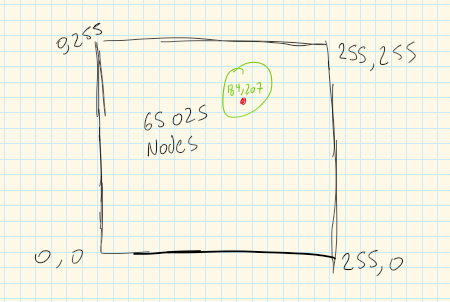


Figure - Particle Location

Now it is necessary to get the index of the particle, the reason lies on the fact that this index is going to be later used for finding the particle position on the map passed in the parameters of the Erode method. To be able to get the index the author used linear interpolation [Figure 12].

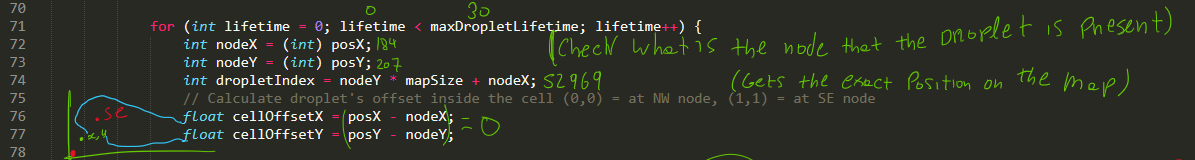


Figure - Example of a water particle creation

###### 3)Calculate and update the direction for the water particle using the gradient

For calculating the gradient for the particle of water the method named CalculateGradient() was used [Figure 13]. This method returns a value of type gradient; this value is defined inside the class as an internal struct to hold three values; height, x position and y position.

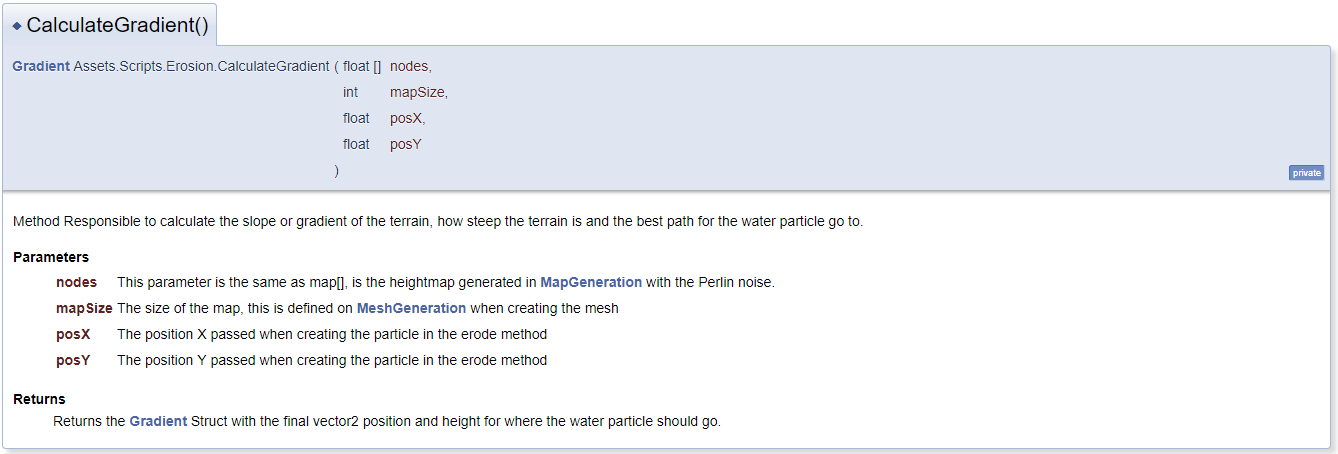


Figure - Calculate Gradient Method

Beyer (2015) sets the new particle direction using the following formula:

In [Figure 14] it is possible to see how the author is updating the particle direction every loop. The reason why the author does not use inertia lies on the fact that he decided to implement it in a different way.

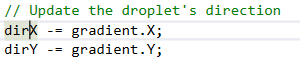


Figure - Update Particle Direction

The next step is to get the length between the x and y for normalizing the vector using the Pythagorean theorem [Figure 15].



Figure - Getting the length of the two positions ||V||

The last step is to get the normalized value from the dirX and dirY (Khan Academy 2019) [Figure 16]. When the value is normalized the author adds it to the position of the water particle.

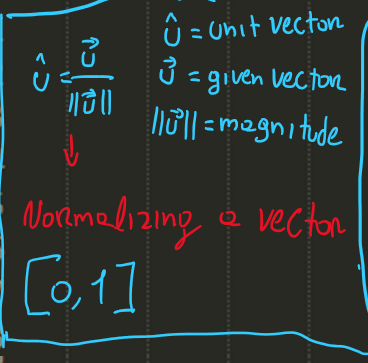


Figure - How to normalize a vector

###### 4) Calculate the new particle’s height

When the particle changes position, it is necessary to get the new current height where it is located. To do this the CalculateGradient method is called, storing the new height to a float variable [Figure 17].



Figure - New Height

The last step is to calculate the delta height between the old height and the new height. Delta in mathematics is simple the difference between a value and the other (Study.com 2019).

###### 5) Calculate the particle sediment capacity

For calculating the sediment capacity, the formula below was used (Volynskov 2011).

Formula

* V = Water velocity
* W = Water amount
* Ch = constant value

###### 6) Erosion and deposition

Now that all the essentials are created the last part is to erode the map itself.

Erosion

When implementing the erosion, a certain amount to be eroded needs to be calculated. To do this it was necessary to set a minimum amount to erode [Figure 18].



Figure - Amount to erode

After this the author wanted to implement a type of inertia, he wanted to create erosions with different sizes to give some randomness in the map.

To do this the author created a float variable named weighedErodeAmount, and set it equals the amounToErode multiplied by a Random Range between 0.01f and Radius. Wit this, is possible to see a small but noticeable difference in the [Figure 19].

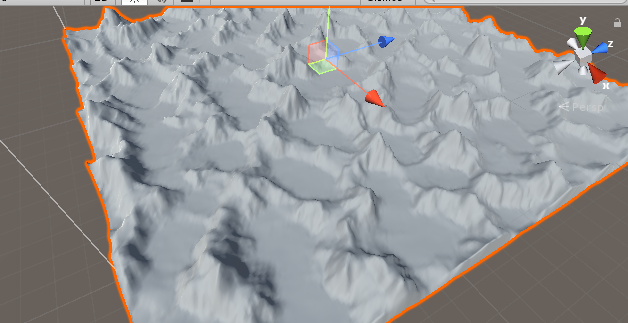
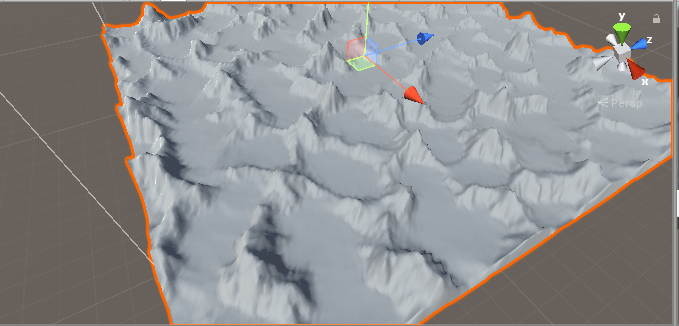


Figure - Before || After

Deposition

For this the first thing is to check if the sediment in the particle is bigger than the sediment capacity. If this turns to be true, then the amount to deposit equals to the sediment in the particle minus the sediment capacity.

##### Gradient Method

To implement the gradient method the first thing was studying how to do linear interpolation, bilinear interpolation and contour maps [Figure 18].

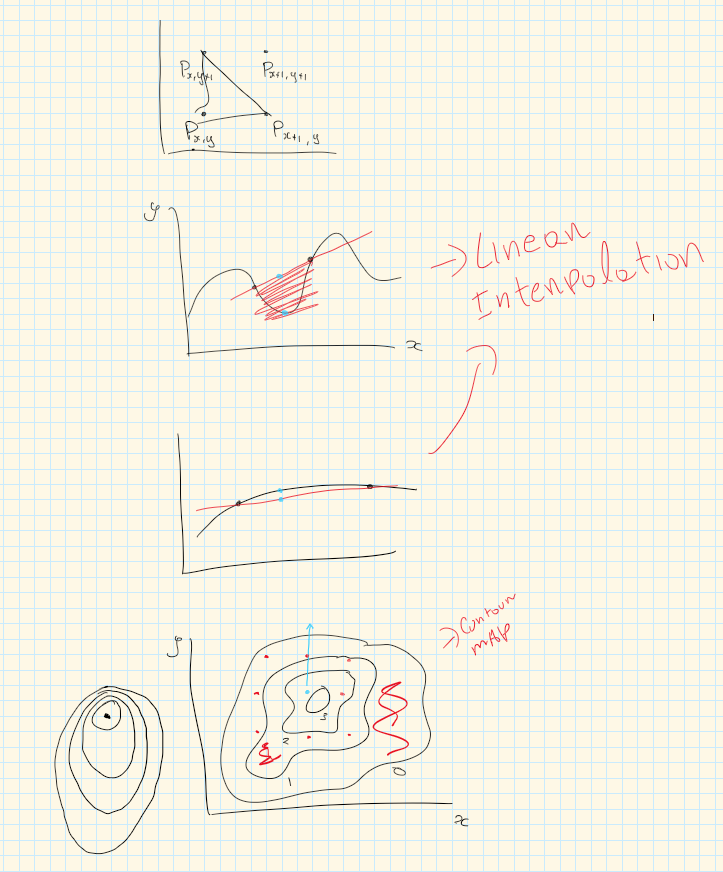


Figure - Droplet gradient calculation sketch

For calculating the gradient, the process was straightforward, check the images bellow.

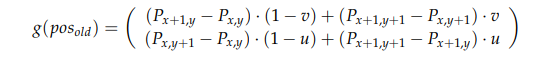


Figure - Beyer, H., 2015. Calculating a gradient using bilinear interpolation [digital image] [viewed 22 March 2019]. Available from:

<https://www.firespark.de/resources/downloads/implementation%20of%20a%20methode%20for%20hydraulic%20erosion.pdf>

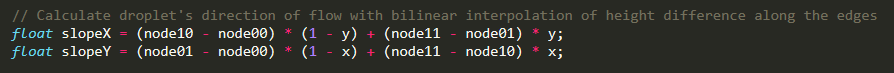


Figure - Bilinear interpolation in code

Untitled picture.png 
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings


Figure - Calculating the four nodes of the particle.

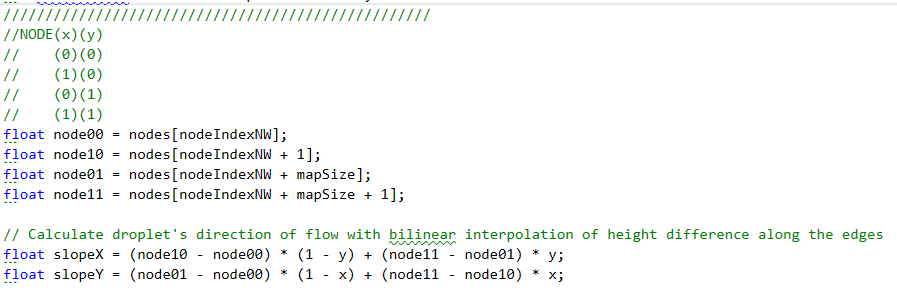


Figure - Gradient Implementation in Code

##### UML Erosion Class Design

A screenshot of a cell phone

Description automatically generated

Figure - UML Erosion Class

### Sprint Review

This sprint not only gave the author a great knowledge in how to implement an erosion algorithm and modifying the mesh to apply the erosion but also improved his mathematics knowledge.

### WBS

1. Research (60%) (36 hours)
2. Fractal Generation (40%) (24 hours)

### References

BEYER, H., 2015. *Implementation of a method for hydraulic  
erosion*, TECHNISCHE UNIVERSITÄT MÜNCHEN

FAIRBRIDGE, R., 1968. Sheet erosion, sheetwash, rainwash, sheetflood. *Encyclopedia of Earth Science,*

FOOD AND AGRICULTURE ORGANIZATION, 1978. *Soil erosion by water.*Repr. ed. Rome: FAO

KHAN ACADEMY, 2019. *Vector magnitude and normalization*[viewed 08/04/ 2019]. Available from: <https://www.khanacademy.org/computing/computer-programming/programming-natural-simulations/programming-vectors/a/vector-magnitude-normalization?fbclid=IwAR2-EO05GIcGXZ0b3o8eoQhL4RwQ1J_W839b3hxFaDW97tUbtO_pfDX7gTU>

KLAUSMEYER, K., 2015. Streambank Erosion.

LEOPOLD, L.B., M.G. WOLMAN and J.P. MILLER, 1995. *Fluvial processes in geomorphology.*New York: Dover Publ

MICROSOFT .NET, 2018. *Random Class*[viewed 08/04/ 2019]. Available from: <https://docs.microsoft.com/en-us/dotnet/api/system.random?view=netframework-4.8>

RECKENDORF, F., 2010. *STREAMBANK EROSION CAUSES,  
AND  
LARGE WOODY MATERAL (LWM)  
SOLUTIONS FOR  
STREAMBANK EROSION AND SEDIMENT REDUCTION*2nd Joint Federal Interagency Conference, Las Vegas: [viewed 20/04/2019]. Available from: <https://acwi.gov/sos/pubs/2ndJFIC/Contents/11D_Reckendorf_02_23_10.pdf>

VOLYNSKOV, A., 2011. *Water erosion on heightmap terrain*[viewed 06/04/ 2019].

### Bibliography

BEYER, H., 2015. *Implementation of a method for hydraulic  
erosion*, TECHNISCHE UNIVERSITÄT MÜNCHEN

DOCS MICROSOFT, 2015a. *Jagged Arrays (C# Programming Guide)*[viewed 08/04/ 2019]. Available from: <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/arrays/jagged-arrays>

DOCS MICROSOFT, 2015b. *Single-Dimensional Arrays (C# Programming Guide)*[viewed 08/04/ 2019]. Available from: <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/arrays/single-dimensional-arrays>

FAIRBRIDGE, R., 1968. Sheet erosion, sheetwash, rainwash, sheetflood. *Encyclopedia of Earth Science,*

FOOD AND AGRICULTURE ORGANIZATION, 1978. *Soil erosion by water.*Repr. ed. Rome: FAO

KHAN ACADEMY, 2019. *Vector magnitude and normalization*[viewed 08/04/ 2019]. Available from: <https://www.khanacademy.org/computing/computer-programming/programming-natural-simulations/programming-vectors/a/vector-magnitude-normalization?fbclid=IwAR2-EO05GIcGXZ0b3o8eoQhL4RwQ1J_W839b3hxFaDW97tUbtO_pfDX7gTU>

KLAUSMEYER, K., 2015. Streambank Erosion.

LEOPOLD, L.B., M.G. WOLMAN and J.P. MILLER, 1995. *Fluvial processes in geomorphology.*New York: Dover Publ

MDN WEB DOCS, 2019. *Math​.max()*[viewed 15/05/ 2019]. Available from: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math/max>

MICROSOFT .NET, 2018. *Random Class*[viewed 08/04/ 2019]. Available from: <https://docs.microsoft.com/en-us/dotnet/api/system.random?view=netframework-4.8>

MICROSOFT DOCS, 2019. *Arithmetic operators (C# Reference)*[viewed 15/04/ 2019]. Available from: <https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/operators/arithmetic-operators#remainder-operator->

OLSE, J., 2004. Realtime Procedural Terrain Generation. *Department of Mathematics And Computer Science, University of Southern Denmark,*

PERSSON, M., 2010. *The Word of Notch*[viewed 15/04/ 2019]. Available from: <https://notch.tumblr.com/post/1409584546/some-work-on-biomes>

POLLANO, F., 2012. *Calculating the distance between 2 points*[viewed 07/04/ 2019]. Available from: <https://stackoverflow.com/questions/11555355/calculating-the-distance-between-2-points?fbclid=IwAR2ih_elI7nYonLhZ8zp2WVyDSUBMztaFtcorOe4toXT0NuJZZgrZu4uyKY>

RECKENDORF, F., 2010. *STREAMBANK EROSION CAUSES,  
AND  
LARGE WOODY MATERAL (LWM)  
SOLUTIONS FOR  
STREAMBANK EROSION AND SEDIMENT REDUCTION*2nd Joint Federal Interagency Conference, Las Vegas: [viewed 20/04/2019]. Available from: <https://acwi.gov/sos/pubs/2ndJFIC/Contents/11D_Reckendorf_02_23_10.pdf>

RED BLOB GAMES, 2015. *Making maps with noise functions&nbsp;*[viewed 09/04/ 2019]. Available from: <https://www.redblobgames.com/maps/terrain-from-noise/#trees>

ŠT'AVA, O.*et al.,* Jul 7, 2008. Interactive terrain modeling using hydraulic erosion. Eurographics Association, pp.201-210

STUDY.COM, 2019. *What is Delta? - Definition &amp; Concept*[viewed 20/04/ 2019]. Available from: <https://study.com/academy/lesson/what-is-delta-definition-lesson-quiz.html>

VOLYNSKOV, A., 2011. *Water erosion on heightmap terrain*[viewed 06/04/ 2019].

WIKIPEDIA, 2018. *Fractional Brownian motion*[viewed 09/04/ 2019]. Available from: <https://en.wikipedia.org/wiki/Fractional_Brownian_motion>

WIKIPEDIA, 2019. *Contour line*[viewed 06/04/ 2019]. Available from: <https://en.wikipedia.org/wiki/Contour_line>

YOUNG, M.*, et al.*, 2004. *Euclidean and Euclidean Squared*2019]. Available from: <http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Clustering_Parameters/Euclidean_and_Euclidean_Squared_Distance_Metrics.htm?fbclid=IwAR3tWg3O_7vWxwPjEzIh3PgmskyoE1PLcYJMNiqHBk741JvO5C7BExpEc2M>