ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2024
M. Skouras and H. Wang
(Guest Editors)

Volume 43 (2024), Number 8

Underwater terrain generation

Abstract

In this paper we present a novel method to generate underwater landscapes adaptable for large and small scale terrains by using a accumulative generation. The creation and evolution of elements of the terrain is managed by the use of environmental objects.

CCS Concepts

• ***Computing methodologies*** → *Collision detection;* • ***Hardware*** → *Sensors and actuators; PCB design and layout;*

1. Introduction

- Terrain generation
Automated terrain generation is a key component of natural scene digital modeling for animated movies and video games. Many landscapes have been studied and are synthetised with more and more realism. Different processes can be used and combined to achieve these scenes: fractal terrains, erosion simulation, hand modeling, geological simulation, ... [REFS] The high quality of synthesis for such environment is due to the possibility to observe these environments form many point of views: long-distance gazing, hiking on mountains, remote sensing, aerial imaging, ... [REFS?]. Thanks to the quality of the digital modeling, the entertainment industry display often breathtaking land scenes.

- Absence of underwater generation
Underwater scenes are rarely created in these media for multiple reasons: these environments are not completely understood and mastered as much as land environments because they are difficult to access, we lack the capacity to see them at a larger scale (unlike mountains for example) and the underlying process that forms these landscapes are much more complex to simulate [REF TERRESTRIAL SIMULATION / UNDERWATER SIMULATION].

These limitations cause animated movies and video games studios to avoid as much as possible underwater environments.

- Needs for robotics and biology
However, these environments are important for the study of biology, geology, and, by extension, robotics. Due to the complexity of an underwater human expedition, autonomous underwater vehicles (AUV) begins to be used for these studies. Validation of the navigation systems require simulations, but roboticians are lacking the capacity to test the robot's algorithms on realistic virtual scenes, and so only test them on synthetic scenarios that do not correlate with real world terrains.

- Lack of multi-scale models
The difficulties to visualize and study the underwater environments on a large scale at the same time as the small scale is an obstacle to the generation and simulation of scenes that are coherent in these two different scales.

- Long time for simulations
A solution to this problem is to simulate at the smallest scale the behaviour of all the elements of the environment. Computing such a simulation in order to generate a full island is an impossible task due to time and memory complexity.

- Objects definition
The method we propose provides a way to generate environments on multiple scales without introducing such complexity by using a sparse representation of the environment using environmental objects combined with cellular automata to link all the elements of the terrain together. The environmental objects only have access to local properties of the environments to spawn, grow and die. The use of local interaction with global properties removes the need for complex interconnections between all elements of the terrain, providing a parallelisable generation process at large to small scales. Defining objects of the terrain as parametric models based on point, curve or region skeletons provides a lightweight representation of the terrain. Our method does not aim for a visually realistic generation, but for a geological plausible terrain. Using state of the art modeling of the geometry of the elements of the terrain could achieve realistic results.

2. Related works

- Landscape generation [GGP*19]
– Blabla fractal terrains [MKM89]
– Blabla data driven [KGG*20] / learning [BSS07] / Markov chain
– Blabla erosion simulation [CJP*23, SPF*23, PPG*19, **?**]
– Blabla subsurface geology [PNL*21] / tectonic simulation [CPEGEG19, **?**]
– Blabla sketch based [EGDEG*17, TB18]

- Cellular automata

- Environmental objects [GPEG*16, EGDEGP16]

- Aerodynamic animation [WH91]

3. Method

The overall pipeline of the method is based on simple incremental generation [FIND REF USING THIS PIPELINE].

3.1. Pipeline

- Initial terrain
The generation of the terrain is initialized using an initial height field. Different materials can be defined, including properties such as the diffusion speed, the mass, the damping factor [and the influence from the water currents]. Finally, a list of environmental objects are provided with their properties: type, size, generation rules, growing conditions, absorption and deposition rate and effects on the water currents.

- Generation loop
The generation is incremental and the main loop is composed of two different steps: the instantiation of objects and the update of the environment.

- Object instantiation
At each time step, an element of the terrain is created at its most fitting location. [STILL NEED TO FIND HOW TO CHOOSE THE GOOD OBJECT]. Once the best fitted object is instantiated at the best fitted location, the process can continue.

- Environment update
At each time step, the environment's properties are updated using a simple cellular automaton. We consider the materials available in the terrain to . Taking into account the water currents and the heightmap's slope, the material is displaced using a distorsion field $D_{\mathcal{M}}$ defined as:

$$D_{\mathcal{M}} = (W \cdot W_{\mathcal{M}} + \Delta h \cdot m_{\mathcal{M}})\, dt$$

with W the water velocity, $W_{\mathcal{M}}$ the influence of water on the material $\mathcal{M}$, Δh the slope of the terrain and $m_{\mathcal{M}}$ the mass of material.

The amount of material at each point p

- Output result

3.2. Environmental objects

3.2.1. Generation rules

Generation rules provides a cost function defining the best location for an object to spawn. Cost function's parameters contains, for every point p the location to closest objects, the amout of material available and the velocity of the water currents. Additional information about surrounding objects are the curvature and the signed distance from the curve defining curve- and area-based objects, and start and end points of the curve-based objects.

3.2.2. Point, curve and area generation

The seed point of a spawning object is defined by a stochastic process by selecting multiple candiate points in the plane. The gradient of the cost function is used to move the candidates a few iterations; the candidate pinning to a location with maximal score is kept as a seed [MAYBE KEEP AND EXTEND THAT, OR REMOVE ENTIRELY]. For a point-based object, this is all. For a curve- or region-based objects, we need to define the curve that will define it. Region-based objects are defined by the isolevel on which the seed point is pointing. Curve-based objects are defined by a starting point S and ending point E defined at $(S,E) = \left(p - \frac{1}{2}\frac{l\nabla f(p)}{||\nabla f(p)||}, p + \frac{1}{2}\frac{l\nabla f(p)}{||\nabla f(p)||}\right)$ with l the optimal length of the curve.

3.3. Communication between objects and environment

3.3.1. Water currents

- Definition and approximation of currents
- Modification of the currents

3.3.2. Environment materials

- Material deposition
- Material dispersion
- Interaction between materials
- Interaction with terrain surface

4. Results

4.1. Small-scale

- Coral colonies fighting

4.2. Mid-scale

- Canyon scene

4.3. Large-scale

- Coral island scene

5. Discussion

- Limits

- Future
We limited our work to the use of height fields as they are more easily interpretable and more light. However, we want to continue this work by using 3D representations of the terrain and the environment. The different possibilities to explore for this would be: the use of 3D particles to represent the state of the materials in the environment, or voxel grids or flatten representation of the terrain's

surface (but would not allow a different morphological shape than the heightfield...).

In this work we simplify and consider the physics and the exchanges between the objects such that at each generation step the system is in a stable state. While this is wrong in regard to the dynamic system of an ecosystem, we consider that the advantage of this assumption is sufficient. Since we suppose a stable system, we could reduce the complexity of our generation by removing the material motion and represent the problem in a form of an oriented spatialized graph. Each node of the graph represent objects and each oriented edge is a material exchange. Using the analytic form of the water current representation, we do not need to use any simulation steps anymore. More indepth studies could be done in this direction to achieve multi-scale underwater generation without computation overhead.

6. Conclusion

References

[BSS07] BROSZ J., SAMAVATI F. F., SOUSA M. C.: Terrain synthesis by-example. *Communications in Computer and Information Science 4 CCIS* (2007), 58–77. URL: , doi:10.1007/978-3-540-75274-5_4.

[CJP*23] CORDONNIER G., JOUVET G., PEYTAVIE A., BRAUN J., CANI M.-P., BENES B., GALIN E., GUÉRIN E., GAIN J.: Forming terrains by glacial erosion. *ACM Transactions on Graphics 42* (2023), 14. URL: , doi:10.1145/3592422\"{\i}.

[CPEGEG19] CORTIAL Y., PEYTAVIE A., ÉRIC GALIN, ÉRIC GUÉRIN: Procedural tectonic planets. *Eurographics Computer Graphics Forum 38* (2019), 1–11. doi:10.1111/cgf.13614.

[EGDEG*17] ÉRIC GUÉRIN, DIGNE J., ÉRIC GALIN, PEYTAVIE A., WOLF C., BENEŠ B., MARTINEZ B.: Interactive example-based terrain authoring with conditional generative adversarial networks. *ACM Transactions on Graphics 36* (2017). doi:10.1145/3130800.3130804.

[EGDEGP16] ÉRIC GUÉRIN, DIGNE J., ÉRIC GALIN, PEYTAVIE A.: Sparse representation of terrains for procedural modeling. *Computer Graphics Forum 35* (2016), 177–187. doi:10.1111/cgf.12821.

[GGP*19] GALIN E., GUÉRIN E., PEYTAVIE A., CORDONNIER G., CANI M.-P., BENES B., GAIN J.: A review of digital terrain modeling. *Computer Graphics Forum 38* (5 2019), 553–577. URL: , doi:10.1111/cgf.13657.

[GPEG*16] GROSBELLET F., PEYTAVIE A., ÉRIC GUÉRIN, ÉRIC GALIN, MÉRILLOU S., BENES B.: Environmental objects for authoring procedural scenes. *Computer Graphics Forum 35* (2 2016), 296–308. URL: , doi:10.1111/cgf.12726.

[KGG*20] KAPP K., GAIN J., GUÉRIN E., GALIN E., PEYTAVIE A.: Data-driven authoring of large-scale ecosystems. *ACM Transactions on Graphics 39* (11 2020). doi:10.1145/3414685.3417848.

[MKM89] MUSGRAVE F. K., KOLB C. E., MACE R. S.: The synthesis and rendering of eroded fractal terrains. *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1989* (1989), 41–50. URL: , doi:10.1145/74333.74337.

[PNL*21] PATEL D., NATALI M., LIDAL E. M., PARULEK J., BRAZIL E. V., VIOLA I.: Modeling terrains and subsurface geology. *Interactive Data Processing and 3D Visualization of the Solid Earth* (2021), 1–43. URL: , doi:10.1007/978-3-030-90716-7_1.

[PPG*19] PARIS A., PEYTAVIE A., GUÉRIN E., ARGUDO O., GALIN E.: Desertscape simulation. *Computer Graphics Forum 38* (10 2019), 47–55. URL: , doi:10.1111/cgf.13815.

[SPF*23] SCHOTT H., PARIS A., FOURNIER L., GUÉRIN E., GALIN E.: Large-scale terrain authoring through interactive erosion simulation. *ACM Transactions on Graphics 42* (10 2023), 1–15. URL: , doi:10.1145/3592787.

[TB18] TALGORN F. X., BELHADJ F.: Real-time sketch-based terrain generation. *ACM International Conference Proceeding Series* (2018), 13–18. URL: , doi:10.1145/3208159.3208184.

[WH91] WEJCHERT J., HAUMANN D.: Animation aerodynamics. *Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1991 25* (1991), 19–22. URL: , doi:10.1145/122718.122719.