

Aesthetic Terrain Programs Database for Creativity Assessment

Miguel Frade, F. Fernandez de Vega and Carlos Cotta

Abstract—To speed up content creation video game industry is increasingly turning to procedural content generation methods. However, creating and fine tuning procedural algorithms is a time consuming task. To address this issue several Search-Based Procedural Content Generation (SBPCG) techniques have been devised. They propose to automatically search the right input parameters or to generate the procedure itself to produce content with the desired characteristics. In this paper we present a database with 17 820 procedures generated by Genetic Terrain Programming (GTP), a SBPCG technique. Those procedures were evolved using the weighted sum of two morphological metrics to generate terrains with aesthetic appeal for video games. With this database we aim to establish a comparison base for future research regarding creativity of GTP and aesthetic terrains diversity.

I. INTRODUCTION

The video game industry is increasingly turning to procedural content generation methods to automate content creation [1]. However, coming up with good results from a procedure often degenerates into an authoring process of trial and error. Procedural algorithms present a certain degree of unpredictability, so designers might end up performing a lot of tests and simulations until they learn how the procedural system behaves to tune it [2]. The search for the right input parameters and algorithm tune is time consuming. To address this issue several Search-Based Procedural Content Generation (SBPCG) techniques [3] have been devised. Techniques in this category propose to automatically search the right input parameters or to generate the procedure itself that will produce content with the desired characteristics.

One kind of video game content that can take advantage of SBPCG techniques is terrain. For instance, Stachniak and Stuerzlinge [4] use a stochastic local search algorithm that finds an acceptable set of deformation operations to apply to a base terrain in order to obtain a map that approximately adheres to the specified constraints. An evolutionary approach was proposed by Ong et al. [5], where genetic algorithms are used to transform height maps in order to conform them to the required features. The 2D terrain silhouette and a database of representative height map samples are the only form of control. Ashlock et al. [6] propose co-evolution of L-systems parameters and grammar to fit a specific terrain shape, which has some resemblance to symbolic regression. A different perspective is proposed by Togelius et al. [7]. They apply multi-objective EAs to evolve height maps that fit some user

predicted entertainment metrics to hopefully increase players interest on the game. This concept is further developed and applied to StarCraft video game [8]. However, none of these approaches addresses aesthetic appeal or creativity of the generated terrains. Automated fitness assignment based on aesthetic measures is new and emerging field of research, some examples of evolutionary art systems that use automated fitness assignment to images can be found in [9], [10].

We developed Automated Genetic Terrain Programming (GTP_a), a SBPCG technique, to generate procedural terrains for video games. GTP_a utilizes Genetic Programming (GP) as an automated evolutionary search tool for procedural terrains, designated Terrain Programs (TPs). This approach allows the generation of new terrain types with aesthetic appeal. However, unlike other evolutionary techniques where aesthetic evaluation is performed by humans, our technique relies only on geomorphological metrics. Those metrics are accessibility score [11] and obstacles edge length score [12]. The evolutionary search of terrains with these metrics produces TPs that do not require any parameter input to control its look. Therefore, TPs can be integrated in video games without a human performing parameter tuning, thus allowing to save time.

Although we have successfully tested the technique in previous papers [11], [12], the question about GTP_a creativeness remains open. One of the most influential research on how to assess software creativity comes from Ritchie [13]. He proposes a set of criteria to assess programs' creativity based on the artifacts they produce. Pereira et al. [14] apply Ritchie's criteria to a set of systems and suggest also that if a program repeats itself later on it is a sign of less creativity. Still, Colton [15] argues that creativity assessment based only on produced artifacts is not enough. He suggests that creativity assessment should account also for the process the software performs and assess its functionality. In this paper we present a database of TPs (the artifacts) that are the result of a series of tests performed with GTP_a with both accessibility and obstacles edge length metrics combined. We aim to establish a comparison base for future research regarding creativity of GTP_a and aesthetic terrain diversity.

Section II details the GTP_a technique and the used geomorphological metrics. The applied test methodology is presented in Section III, the resulting TPs and preliminary analysis are discussed in Section IV. Some sample terrains are displayed in Section V and finally, conclusions and future work are laid out in Section VI.

Miguel Frade Escola Superior de Tecnologia e Gestão, Instituto Politécnico de Leiria, Portugal; email: miguel.frade@ipleiria.pt

F. Fernandez de Vega Centro Universitario de Mérida, Universidad de Extremadura, Spain email: fcofdez@unex.es

Carlos Cotta ETSI Informática, Campus de Teatinos, Universidad de Málaga, Spain email: ccottap@lcc.uma.es

TABLE I
GP FUNCTION SET

Name	Description
$plus(a, b)$, $minus(a, b)$, $multiply(a, b)$	arithmetical functions
$sin(a)$, $cos(a)$, $tan(a)$, $atan(a)$	trigonometric functions
$exp(a)$	returns e^a
$myLog(a)$	returns 0 if $a = 0$ and $log(a)$ otherwise
$myPower(a, b)$	returns 1 if $b = 0$, 0 if $a = 0$ and $ a ^b$ otherwise
$myDivide(a, b)$	returns a if $b = 0$ and $a \div b$ otherwise
$mySqrt(a)$	returns $\sqrt{ a }$
$negative(a)$	returns $-a$

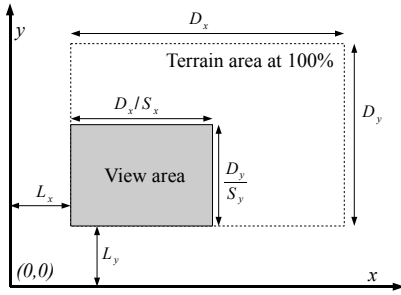


Fig. 1. Terrain view area

II. BACKGROUND

GTP_a , our SBPCG technique, consists on the application of GP [16] to evolve mathematical expressions, designated TPs, that will generate artificial terrains. The function set used in GTP_a is described in Table I and three different terminal sets were used during our tests: $T_1 = \{myNoise(x, y), ERC\}$, $T_2 = \{X, Y, ERC\}$ and $T_3 = \{myNoise(x, y), X, Y, ERC\}$. ERC stands for *Ephemeral Random Constant* [16] and $ERC \in [0, 10]$. The terminal $myNoise(x, y)$ is a lattice noise function based on Blender 3D lattice noise primitive $orgBlenderNoise^1$, see Eq. (1). The ideal properties and mathematical details of a noise function are presented by [17], [18].

$$myNoise(x, y) = 2 \times orgBlenderNoise(x, y, 0) - 1 \quad (1)$$

To evaluate TPs it is required to convert them to an height map. The altitude values h , where $h = f(x, y)$, $h, x, y \in \mathbb{R}$, are stored in matrix $H = \{h_{r,c}\}_{\substack{r \leq n_r \\ c \leq n_c}}$, whose size $n_r \times n_c$ defines the height map resolution. Equation (2) shows the relationship between the height map matrix H and TPs continuous functions. The value $h_{r,c}$ represents the elevation value for row r and column c , and D_x, D_y are the terrain dimensions. S_x, S_y allow the control of the zoom level and L_x, L_y allow us to localize the origin of the terrain view area (see Fig. 1).

¹Source code available at <http://www.blender.org/>

$$h_{r,c} = f \left(\frac{c \times \frac{D_x}{n_c - 1}}{S_x} + L_x, \frac{r \times \frac{D_y}{n_r - 1}}{S_y} + L_y \right) \quad (2)$$

Tps are evaluated by the weighted sum of two morphological metrics: accessibility score [11] and obstacles edge length score [12]. The accessibility score aims to generate terrains where a certain percentage of the terrain area is accessible, which means, its slope is under a certain threshold. Therefore, the slope map $S = \{s_{r,c}\}_{\substack{r \leq n_r \\ c \leq n_c}}$ is created to store the declination for each cell r, c of the height map H . The slope values are calculated using B. Horn [19] method. Then, to determine the cells that are accessible the accessibility map $A = \{a_{r,c}\}_{\substack{r \leq n_r \\ c \leq n_c}}$ is created, with either 0 (not accessible) or 1 (accessible) in each cell depending on the selected slope threshold. To allow player units to move around the accessible cells should be connected in an large area, which is determined by a component labeling algorithm. Finally the accessibility score is determined by Eq. (3), where $p_a \in [0, 1]$ is a threshold to avoid the appearance of completely flat terrains and represents the percentage of desired accessible area. More details about this metric can be found in [11].

$$v_s = \left| \frac{n_r n_c}{\sum_{r=1}^{n_r} \sum_{c=1}^{n_c} a_{r,c}} - \frac{n_r n_c}{\lceil p_a n_r n_c \rceil} \right|, \quad p_a \neq 0 \quad (3)$$

The edge length score aims to increase the amount of terrain obstacles and its edge complexity [12]. To calculate this score the binary edge map $E = \{e_{r,c}\}_{\substack{r \leq n_r \\ c \leq n_c}}$ must be created. This is achieved with the Laplacian operator [20], which returns a positive value when a cell $a_{r,c}$ belongs to the edge line and the correspondent cell $e_{r,c}$, is filled with value 1. Based on the amount of cells that belong to the edge, we classify the terrain by the edge score ε_s defined in Eq. (4), where $p_e \in [0, 1]$ is a threshold to allow the formation of large accessible areas and represents the desired percentage of edge length in relation to the total terrain area. More details about this metric can be found in [12].

$$\varepsilon_s = \left| \frac{n_r n_c}{\sum_{r=1}^{n_r} \sum_{c=1}^{n_c} e_{r,c}} - \frac{n_r n_c}{\lceil p_e n_r n_c \rceil} \right|, \quad p_e \neq 0 \quad (4)$$

The fitness function to evaluate TPs is the weighted sum of the two previous metrics, as shown in Eq. (5), where $w_a + w_e = 1$ (due to this relation, from now on, we will refer only to w_a).

$$fitness = w_a v_s + w_e \varepsilon_s \quad (5)$$

Although the GTP_a technique has been presented in previous publications [11], [12], only a small amount of TPs have been generated and the fitness function included only a single metric. Our aim is to generate a large set of TPs to assess the creativeness of the used technique and serve as base for future comparisons.

TABLE II
TEST PARAMETERS AND THEIR VALUES

Par.	Value	Par.	Value
T_1	{ERC, myNoise}	s_1	18%
T_2	{ERC, X, Y}	s_2	27%
T_3	{ERC, X, Y, myNoise}	s_3	36%
pa_1	70%	pe_1	20%
pa_2	80%	pe_2	25%
pa_3	90%	pe_3	30%
w_a	0.0, 0.1, ..., 1.0	$seed$	$run = \{1, 2, \dots, 20\}$

TABLE III
GP PARAMETERS

GP	Value
maximum generations	50
population size	500
initialization method	half and half
ramped	from 2 to 6
max. depth	17
selection operator	tournament, size 7
crossover operator	rate 70%
mutation operator	subtree, rate 30%

III. TESTS

Evaluation of TPs depends on: slope threshold (from now on represented by s), percentage of accessibility area pa , percentage of the edge length pe and weights w_a and w_e . Table II presents a set of parameters, designated as *Test Parameters*, whose influence we want to study. To assess the creativity and diversity of terrains produced by GTP_a we performed a series of tests that included all combinations between the test parameters T_i , s_j , pa_k , pe_l and w_m . For each combination, 20 runs ($r = 1, 2, \dots, 20$) were performed with different seeds, which sums to 17 820 different executions [21].

Table III presents the used GP parameters, which were fixed during all runs. The search stops whenever the fitness reaches the value of zero or the amount of generations reaches the value of 50, whichever comes first. Both crossover and mutation operators are the same as the ones used by Koza [16]. Table IV shows the Height Map Parameters required to convert TPs to high maps (also fixed across all runs).

IV. DATABASE

To establish a comparison base for future research regarding creativity of GTP_a as well as aesthetic terrains diversity, we decided to release a TPs database. This database, formatted

TABLE IV
HEIGHT MAP PARAMETERS

Height map	Value
n_r and n_c	128
L_x and L_y	0
S_x and S_y	1
D_x and D_y	10

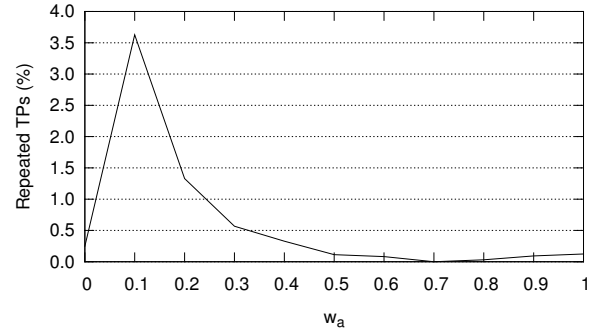


Fig. 2. Percentage of repeated TPs versus w_a .

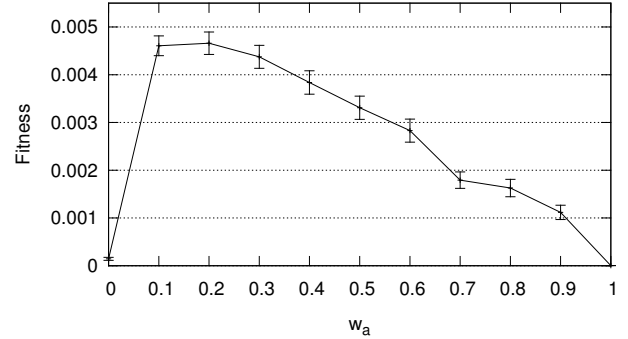


Fig. 3. Average fitness values versus w_a . Error bars represent the standard error of the mean.

as comma separated values (CSV) file, contains the results from our 17 820 different executions, with the following fields: *terminal*; *slope*(%); *pa*(%); *pe*(%); w_a ; *run*; *fitness*; *TP*;. The *fitness* value is standardized, so lower values are better. Although the fitness value does not give any information regarding the creativity of our system, it is included to indicate how feasible/unfeasible a given TP is regarding our metric. The amount of TPs that reached the perfect fitness value (zero) was 45.22%. Below is an example line of the CSV file:

```
T3; 18; 70; 20; 0.0; 09; 0.000000000;
myPower(cos(myNoise(X,Y)),exp(myNoise(X,Y)));
```

Due to the large amount of results, we decided to split our database by terminal set: *TPs_T1.csv*, *TPs_T2.csv* and *TPs_T3.csv*. The database is available to the public in the Sourceforge repository <http://sourceforge.net/p/tps-db/> under the Creative Commons Attribution-ShareAlike 3.0 Unported License². We have also added to the repository some C code to show how to calculate the height values from TPs.

One of our goals is to find diverse solutions, which can also be considered a way to assess the creativity of GTP_a (on a limited sense). Pereira et al. [14] suggest that if a system presents repetitions in later runs that is a sign of less creativity. So, in our preliminary assessment of creativity we looked for any repeated TPs in our database. We found 98.61% of unique genotypes and a total of 106 TPs that appeared more than once, relative to 248 runs (1.39%). Figure 2 shows how the repeated

²License available at <http://creativecommons.org/licenses/by-sa/3.0/>

TABLE V

RUNS WHERE TP SHOWN IN EQ. (6) WAS THE BEST SOLUTION AND THEIR CORRESPONDENT FITNESS VALUES.

<i>terminal</i>	<i>slope</i>	<i>pa</i>	<i>pe</i>	w_a	<i>run</i>	<i>fitness</i>
T_1	s18	a80	e20	0.1	18	0.03484
T_1	s18	a90	e20	0.1	18	0.04143
T_3	s18	a80	e20	0.1	6	0.03484
T_3	s18	a80	e20	0.1	7	0.03484
T_3	s18	a80	e20	0.3	7	0.03438
T_3	s18	a90	e20	0.1	6	0.04143
T_3	s18	a90	e20	0.1	7	0.04143
T_3	s18	a90	e20	0.3	7	0.05390

Ts are distributed in relation to w_a . The higher concentration of repeated Ts is where w_a values have worse fitness values (see Fig. 3), specially for $w_a = 0.1$. For $w_a = 0.7$ there are no repeated Ts. On average, the repeated Ts appear as solution of 2.34 runs. Equation (6) shows the worse case, it appeared 8 times, Table V identifies the runs were this particular TP appeared and the correspondent fitness values. We believe that a larger limit of maximum allowed generations can drastically reduce, or even eliminate, the amount of repeated Ts, but more tests are needed to confirm it.

$$TP = \cos(\cos(\operatorname{atan}(\operatorname{atan}(\operatorname{atan}(\operatorname{atan}(\operatorname{myNoise}(x, y))))))) \quad (6)$$

Our repetition analysis only accounts for different terrains genotypes. However, there might exist also different Ts that are mathematically equivalent and render the same terrain. Another important aspect, from the creativity point of view, is how similar (or diverse) are terrains from our database. Although GTP_a may produce many different phenotypes, we do not know if many of them are similar or not. To answer the question of how many diverse terrains types GTP_a is able to generate further analysis must be conducted.

V. SAMPLES

To illustrate the terrains that GTP_a is able to produce, we selected a few different Ts, which are displayed in Fig. 4, 5, 6, 7, 8, and 9. Those figures show a rendered image of a three dimension view point from the terrain. Those renders were performed in Blender 3D without textures to emphasize terrains surface shape. Each figure has the identification of the TP that generated it with the following syntax: *terminal*, *slope*, *pa*, *pe*, w_a and *seed*. For abbreviation proposes we replaced w_a by w_m and *seed* by r_u , where m can take values in the range $m = 0, \dots, 10$ and $u = 1, \dots, 20$.

Given the huge amount of results we have only performed a visual inspection of 100 randomly selected terrains from each terminal set. Our visual analysis showed a clear relation between terminal sets and both terrain looks and diversity. Terminal set T_1 is the one that presented the lower diversity, but also more realistic look. On the other hand, T_2 presented the highest diversity of terrains. However, terrains from this terminal set tend to present geometric patterns and symmetry, which give them a strange look. Although this characteristic

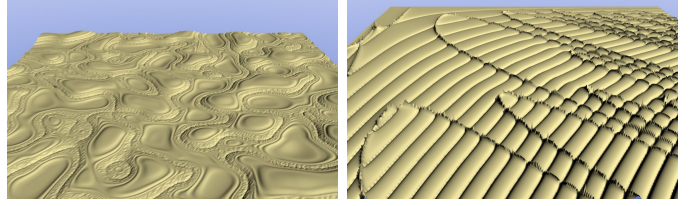


Fig. 4. Terrains generated by TP $T_1, s_1, pa_1, pe_3, w_0, r_4$ with *fitness* = 0.000000 on the left, and $T_2, s_3, pa_2, pe_3, w_0, r_2$ with *fitness* = 0.000000 on the right

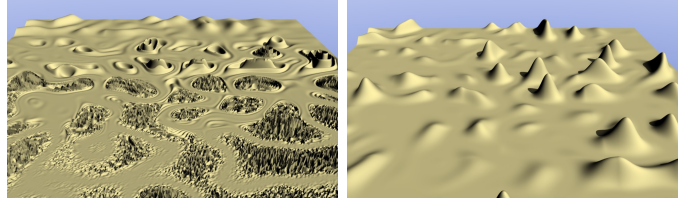


Fig. 5. Terrains generated by TP $T_3, s_3, pa_2, pe_1, w_0, r_{10}$ with *fitness* = 0.000000 on the left, and $T_1, s_3, pa_1, pe_2, w_{10}, r_{10}$ with *fitness* = 0.000000 on the right

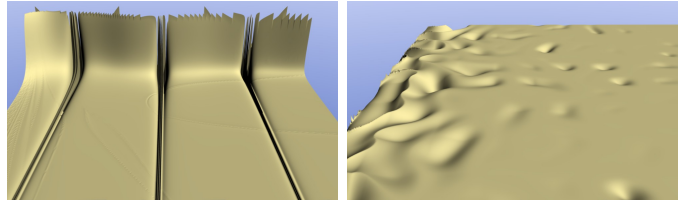


Fig. 6. Terrains generated by TP $T_2, s_2, pa_2, pe_1, w_{10}, r_{19}$ with *fitness* = 0.000000 on the left, and $T_3, s_1, pa_1, pe_2, w_{10}, r_{15}$ with *fitness* = 0.000000 on the right

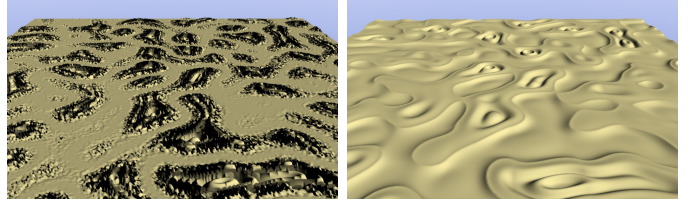


Fig. 7. Terrains generated by TP $T_1, s_2, pa_1, pe_2, w_9, r_9$ with *fitness* = 0.000098 on the left, and $T_1, s_2, pa_2, pe_3, w_8, r_1$ with *fitness* = 0.000000 on the right

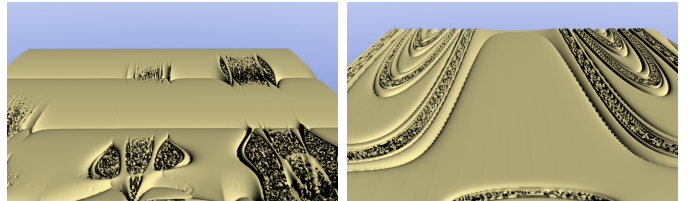


Fig. 8. Terrains generated by TP $T_2, s_2, pa_3, pe_1, w_8, r_3$ with *fitness* = 0.000181 on the left, and $T_2, s_2, pa_3, pe_2, w_9, r_8$ with *fitness* = 0.000068 on the right

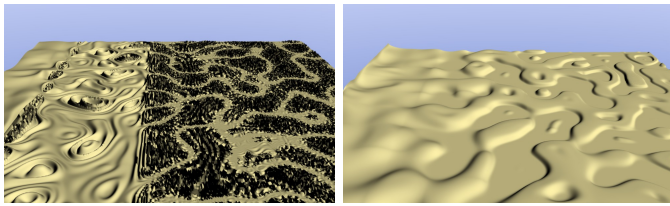


Fig. 9. Terrains generated by TP $T_3, s_3, pa_2, pe_2, w_8, r_{10}$ with $fitness = 0.000000$ on the left, and $T_3, s_3, pa_3, pe_2, w_2, r_8$ with $fitness = 0.020270$ on the right

might be a good indicator of novelty and therefore creativity [13], they might be considered less pleasant. Terrains from terminal set T_3 were the ones that the authors found to be more balanced between aesthetic appeal and diversity.

VI. CONCLUSIONS

A series of tests have been made with GTP_a where the fitness function was the combination of two metrics: accessibility score and edge length score. Our analysis showed that GTP_a is able to find many different solutions that fit our goals, presenting 98.61% of unique solutions. Furthermore, the 1.39% of repetitions were concentrated where the fitness values were worse, which makes us believe that increasing the value of maximum allowed generations would decrease or eliminate them. The visual inspection of some terrains also showed many diverse terrain types, although some of them presented a strange look, specially from terminal T_2 .

So far our creativity analysis on GTP_a is on a preliminary stage and further studies must be conducted, like applying Ritchie's criteria [13]. Another interesting research would be the use of classification system to aggregate terrains by their morphological similarity and this way assess phenotype diversity. This approach poses some challenges on which metric used to classify morphological similarity. A different possibility would be to perform a user study to classify terrain creativity characteristics, like novelty or quality, and its impact on video games replayability. GTP_a evaluates TPs after converting them to height maps, however with this approach if we change the resolution n_r and n_c , their fitness value will likely change. This dependence on the chosen resolution is not desirable, so other approach could be devised to evaluate TPs based on their equations rather than on their phenotype.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions. The authors acknowledge the support of Spanish Ministry of Science and Innovation under project ANYSELF (TIN2011-28627-C04). The first and second authors were also supported by Gobierno de Extremadura, under project GRU09105, GR10029 and Municipality of Almendralejo. The third author was also supported by project TIC-6083 of Junta de Andalucía.

REFERENCES

- [1] M. J. Nelson and M. Mateas, "Towards automated game design," in *AI*IA '07: Proceedings of the 10th Congress of the Italian Association for Artificial Intelligence on AI*IA 2007*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 626–637.
- [2] C. Remo, "MIGS: Far Cry 2's Guay On The Importance Of Procedural Content," Website (accessed on Sep. 2011), 2008, http://www.gamasutra.com/php-bin/news/_index.php?story=21165.
- [3] J. Togelius, G. N. Yannakakis, K. O. Stanley, and C. Browne, "Search-based procedural content generation: A taxonomy and survey," *Computational Intelligence and AI in Games, IEEE Transactions on*, vol. 3, no. 3, pp. 172–186, sept. 2011.
- [4] S. Stachniak and W. Stuerzlinger, "An algorithm for automated fractal terrain deformation," *Computer Graphics and Artificial Intelligence*, vol. 1, pp. 64–76, 2005.
- [5] T. J. Ong, R. Saunders, J. Keyser, and J. J. Leggett, "Terrain generation using genetic algorithms," in *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*. NY, USA: ACM, 2005, pp. 1463–1470.
- [6] D. Ashlock, S. Gent, and K. Bryden, "Embryogenesis of artificial landscapes," in *Design by Evolution*, ser. Natural Computing Series, P. F. Hingston, L. C. Barone, and Z. Michalewicz, Eds. Springer Berlin Heidelberg, 2008, pp. 203–221. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-74111-4_12
- [7] J. Togelius, M. Preuss, and G. N. Yannakakis, "Towards multiobjective procedural map generation," in *PCGames '10: Proceedings of the 2010 Workshop on Procedural Content Generation in Games*. New York, NY, USA: ACM, 2010, pp. 1–8.
- [8] J. Togelius, M. Preuss, N. Beume, S. Wessing, J. Hagelback, and G. Yannakakis, "Multiobjective exploration of the starcraft map space," in *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*. IEEE, 2010, pp. 265–272.
- [9] P. Machado and A. Cardoso, "All the truth about nevar," *Applied Intelligence*, vol. 16, pp. 101–118, 2002, 10.1023/A:1013662402341.
- [10] S. DiPaola and L. Gabora, "Incorporating characteristics of human creativity into an evolutionary art algorithm," *Genetic Programming and Evolvable Machines*, vol. 10, pp. 97–110, 2009, 10.1007/s10710-008-9074-x. [Online]. Available: <http://dx.doi.org/10.1007/s10710-008-9074-x>
- [11] M. Frade, F. Fernández de Vega, and C. Cotta, "Evolution of Artificial Terrains for Video Games Based on Accessibility," in *Applications of Evolutionary Computation*, ser. Lecture Notes in Computer Science, C. D. Chio *et al.*, Eds., vol. 6024. Springer, 2010, pp. 90–99.
- [12] —, "Evolution of Artificial Terrains for Video Games Based on Obstacles Edge Length," in *IEEE Congress on Evolutionary Computation 2010*, Jul. 2010, pp. 1–8.
- [13] G. Ritchie, "Some empirical criteria for attributing creativity to a computer program," *Minds Mach.*, vol. 17, no. 1, pp. 67–99, Mar. 2007. [Online]. Available: <http://dx.doi.org/10.1007/s11023-007-9066-2>
- [14] F. Pereira, M. Mendes, P. Gervás, and A. Cardoso, "Experiments with assessment of creative systems: an application of ritchies criteria," in *Proceedings of the Workshop on Computational Creativity, 19th International Joint Conference on Artificial Intelligence*, vol. 5, 2005, p. 05.
- [15] S. Colton, "Creativity versus the perception of creativity in computational systems," in *Creative Intelligent Systems: Papers from the AAAI Spring Symposium*, 2008, pp. 14–20.
- [16] J. R. Koza, "Genetic Programming. On the programming of computers by means of natural selection," *Cambridge MA: The MIT Press.*, 1992.
- [17] K. Perlin, "An image synthesizer," *SIGGRAPH Comput. Graph.*, vol. 19, no. 3, pp. 287–296, 1985.
- [18] —, "Improving noise," in *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 2002, pp. 681–682.
- [19] B. Horn, "Hill shading and the reflectance map," *Proceedings of the IEEE*, vol. 69, no. 1, pp. 14–47, 1981.
- [20] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Prentice Hall, 2002.
- [21] M. Frade, F. Fernández de Vega, and C. Cotta, "Automatic evolution of programs for procedural generation of terrains for video games," *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, pp. 1–22, 2012, 10.1007/s00500-012-0863-z.