Generation of a random landscape by given configuration entropy and Total Edge

Kuznetsov, Alexander V.
Voronezh State University, Voronezh, 394018, Russia
Contact e-mail: avkuz@bk.ru

The article addresses the development of algorithms for a generation of a random landscape with a given configuration entropy and Total Edge. Two algorithms are proposed. The first one is based on the uniform random filling of a landscape by cells of different types. The second one is based on the probabilistic cellular automaton. The algorithm based on the cellular automaton fills the landscape with cells row by row, and the probability of an "extraordinary" appearance of a new type of cell is predefined. The ratio between cells of different classes is determined from the given configurational entropy. Dependencies between landscape metrics — configuration entropy and Total Edge along with the number of cell types for the landscape built in different ways are demonstrated. Examples of landscapes obtained by the proposed algorithm are shown. These landscape generation algorithms can be used for verification of pathfinding algorithms for the construction of a large number of random landscapes with the same metrics.

Keywords: entropy, Total Edge, landscape generation, cellular automata.

1. Formulation of the problem. Definitions

A configurational entropy of the landscape, Total Edge, and Total Edge Density [1, 2] are widely used characteristics of a landscape (metrics). A configuration entropy describes a quantitative relationship between elements of different classes in a landscape, Total Edge and Total Edge Density metrics characterize to what extent elements of different classes are mixed in a landscape.

These metrics are easy to calculate, and there are a large number of software products for their calculation. However, there is an inverse problem: to generate the landscape $\mathcal{L}(N, l)$ with specified landscape configuration entropy, Total Edge, and Total Edge Density. It is required for testing of algorithms of the pathfinding, such as those described in [3] and for obtaining quantitative characteristics of these algorithms, for example, dependence the efficiency of the algorithm from the landscape features. In this case, a construction of the landscape should be fast enough, as the landscape is necessary, primarily, to accumulate statistics on the time of agents propagating through it. Thus it is needed to sort a large number of different landscapes with the same characteristics within a reasonable computation time.

Of course, one can simply generate random landscapes and then filter those which are suitable for the metrics. However, to obtain a good solution even for a landscape 50 by 50 cells the computation time could exceed several days.

The present work addresses the description of landscape generation algorithms with predefined metrics. Also, paper's goal is to clarify the relationship of these metrics, as in [4] it has been shown that landscapes with the same configuration entropy may look completely different and have very different Total Edge.

We use the cellular automaton for the creation of a landscape with a given Total Edge. Cellular automata are utilized for a landscape generation in computer games. However, the problem to generate a texture or a landscape with characteristics like the given entropy or Total Edge is not commonly stated as it follows from [5, 6].

Definition 1. *The landscape $\mathcal{L}(N,l)$ is the set of N equal-sized cells L_{ij}, $(i,j) \in I \subset \mathbb{N}^2$, belonging to l different classes, where class i belongs to N_i cell, that is $\sum_{i=1}^{l} N_i = N$.*

Cells of landscapes will be divided into classes according to the maximal speed at which these cells can be crossed with in order to test obstacle-avoidance and pathfinding algorithms.

Definition 2. *Let us define a configuration entropy as*

$$S(\mathcal{L}(N,l)) = -\sum_{i=1}^{l} \frac{N_i}{N} \ln \frac{N_i}{N}.$$

Definition 3. *Total Edge (TE) is the total number of the abutting edges of cells belonged to different classes in $\mathcal{L}(N,l)$. We will further denote the Total Edge of the landscape $\mathcal{L}(N,l)$ as $TE(\mathcal{L}(N,l))$.*

Definition 4. *Total Edge Density (TED) of the landscape $\mathcal{L}(N,l)$ is defined as the ratio $TE(\mathcal{L})$ to the total cell quantity in $\mathcal{L}(N,l)$ and would be denoted as*

$$TED(\mathcal{L}(N,l)) = TE(\mathcal{L}(N,l))/N.$$

2. Generation of a landscape with given configuration entropy

In this section, we solve the problem to efficiently generate a landscape $\mathcal{L}(N,l)$ for the given configuration entropy $S(\mathcal{L}(N,l))$. This goal can be achieved by determining a vector $\mathbf{V} = (N_1, \ldots, N_l)$ of numbers N_i of each class i cells that satisfies the following expression

$$\sum_{i=1}^{l} N_i = N, \quad S(\mathbf{V}) = -\sum_{i=1}^{l} \frac{N_i}{N} \ln \frac{N_i}{N} = S(\mathcal{L}(N,l)). \tag{1}$$

We select $N_i = \beta^{i-1} N_1$, $N_1 \leq N$, $\beta \geq 0$. Since interchange of $\mathbf{V}$ components obviously does not change $S(\mathbf{V})$, we consider a non-increasing sequence N_i only, then $0 \leq \beta \leq 1$. If $\beta = 0$, we assume that $\beta^0 = 1$. Then conditions (1) satisfying the known properties of geometric progression can be rewritten as

$$S(\mathbf{V}) = -\sum_{i=1}^{l} \frac{\beta^{i-1} N_1}{N} (\ln \beta^{i-1} + \ln N_1 - \ln N) = -\frac{N_1 \ln \beta}{N} \sum_{i=1}^{l} \beta^{i-1}(i-1) + \ln \frac{N}{N_1}. \tag{2}$$

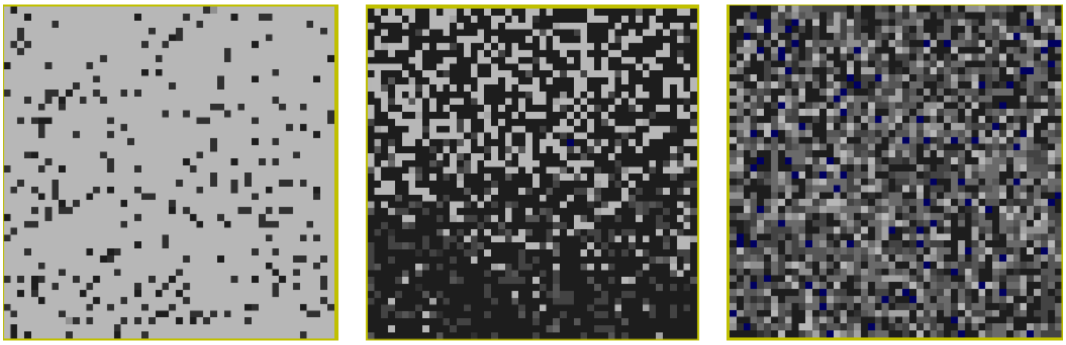

Fig. 1. Landscapes $\mathcal{L}(48 \times 48, 10)$ generated with the algorithm from the section 2. From the left to the right $S = 0.349119$, $TED = 0.40842$, $S = 1.1004$, $TED = 1.05773$, $S = 2.00013$, $TED = 1.63411$

Numbers $\beta^{i-1}(i-1)$ form arithmetico-geometric sequence, therefore

$$N_1 \sum_{i=1}^{l} \beta^{i-1}(i-1) = \frac{\beta}{1-\beta} N_1 \frac{1-\beta^l}{1-\beta} - \frac{N_1 l \beta^l}{1-\beta} = \frac{N\beta - N_1 l \beta^l}{1-\beta}. \quad (3)$$

We substitute (3) into (2) and obtain that

$$S(\mathbf{V}) = -\frac{\beta(1-\beta^l) - (1-\beta) l \beta^l}{(1-\beta)(1-\beta^l)} \ln \beta + \ln \frac{1-\beta^l}{1-\beta}. \quad (4)$$

Thus, the algorithm for the constructing of vectors $\mathbf{V} = (N_1, \ldots, N_l)$ by the given entropy S is the following

1. Solve the equation (4) with respect to β.
2. Define N_1 by the solution found as $N_1 = N(1-\beta)(1-\beta^l)^{-1}$.
3. Construct the vector $\mathbf{V}_0 = (N_1, \beta N_1, \ldots, \beta^{l-1} N_1)$. Round and random shuffle its components to integers and obtain the vector $\mathbf{V}$. It is necessary to trace that the sum of all $\mathbf{V}$ components is equal to the N.

The algorithm for the landscape $\mathcal{L}(N, l) = \{L_{ij}\}$ generation which is based on the vector $\mathbf{V} = (N_1, \ldots, N_l)$ is the following

1. Set the class "-1" to all cells of the $\mathcal{L}(N, l)$.
2. Generate the pair of integers $(i, j) \in I$ from the uniform random distribution.
3. Generate the integer $1 \le k \le l$ from the uniform random distribution so that $N_k \neq 0$.
4. If the class of the L_{ij} is equal to -1, set class of the L_{ij} to k and $N_k := N_k - 1$.
5. Repeat step 2 until $\mathbf{V} \neq \mathbf{0}$.

Examples of landscapes generated with the algorithm described above are shown in Fig. 1. The author has established via the computational experiment that the temporal computational complexity of the algorithm for a rectangular landscape depends on the number of cells N as $O(N^{3/2})$, and on the number of classes of cells l as $O(1)$.

3. Generation of landscape with given Total Edge

Suppose that the landscape $\mathcal{L}(N, l) = \mathcal{L}(N, l) = \{L_{ij} | 1 \le i \le n, 1 \le j \le m\}$. The minimum value of $TE(\mathcal{L}(N, l))$ will be obtained by the landscape $\mathcal{L}(N, l)$ wherein Cl_1 class cells are sequentially placed into one line, then Cl_2 cells etc, starting from one of corners. If cells of one class end, the line of these cells continues by cells of another class. If N_k is the quantity of k-th class cells, then we set $N_k := N_k - 1$ when place a cell of k-th class on the landscape. When the entire line is processed then proceed to the next one, which we process in the opposite direction. Introduce the strict order relation "<" on the lanscape $\mathcal{L}(N, l)$ as follows

$$L_{ij} < L_{sr} = \begin{cases} i < s, \\ j < r, \quad i = s = 2k - 1, \quad 1 \le k \le [n/2] + 1, \\ j > r, \quad i = s = 2k, \quad 1 \le k \le [n/2]. \end{cases}$$

The relation "<" naturally generates operations $\text{pred}(L_{ij}) = L_{sr}$ of the previous cell finding, $L_{sr} < L_{ij}$ and it does not exist L_{pq} so that $L_{sr} < L_{pq} < L_{ij}$, $\text{succ}(L_{ij}) = L_{sr}$ of the successive cell finding, $L_{ij} = \text{pred}(L_{sr})$. Let us denote $Cl(L_{ij})$ as the class of the L_{ij} cell.

The aforementioned algorithm of the landscape filling may be described as the cellular automaton in which the initial state s_{ij} of any cell L_{ij}, $(i, j) \ne (1, 1)$ is equal to -1, the initial state $s_{11} = \xi$, where $\xi \in \overline{1, l}$ is an random class number. The quantity $N(Cl_k)$ of "remaining in reserve" cells of the Cl_k class is defined at each moment of the cellular automaton functioning for each cell class Cl_k. The local transition function is defined as

$$s_{ij}(\varepsilon) = \begin{cases} Cl(\text{pred}(L_{ij})), & Cl(\text{pred}(L_{ij})) > 0, \quad N(Cl(\text{pred}(L_{ij}))) > 0, \; i + j \ne 0, \; \zeta \le 1 - \varepsilon, \\ \eta, & \eta \ne Cl(\text{pred}(L_{ij})) \vee i = n \wedge j = m, \quad \zeta > 1 - \varepsilon, \end{cases}$$
$$N(Cl(s_{ij})) := N(Cl(s_{ij})) - 1,$$

where $\zeta \in [0, 1]$ is an uniformly distributed random number, $\eta \in \overline{1, l}$ is an uniformly distributed random number, $0 \le \varepsilon < 1$.

The rectangular landscape $\mathcal{L}(n \times m, l)$ will have the maximal TE if all cells belong to different classes. The maximum of the configuration entropy for the such landscape will be $S_{\max} = S_{\max}(\mathcal{L}(n \times m, l)) = \ln l$. Numerical experiment (see Fig. 2), which was repeated 50 times for each value of parameters, showed that the average value of the TED depends on ε

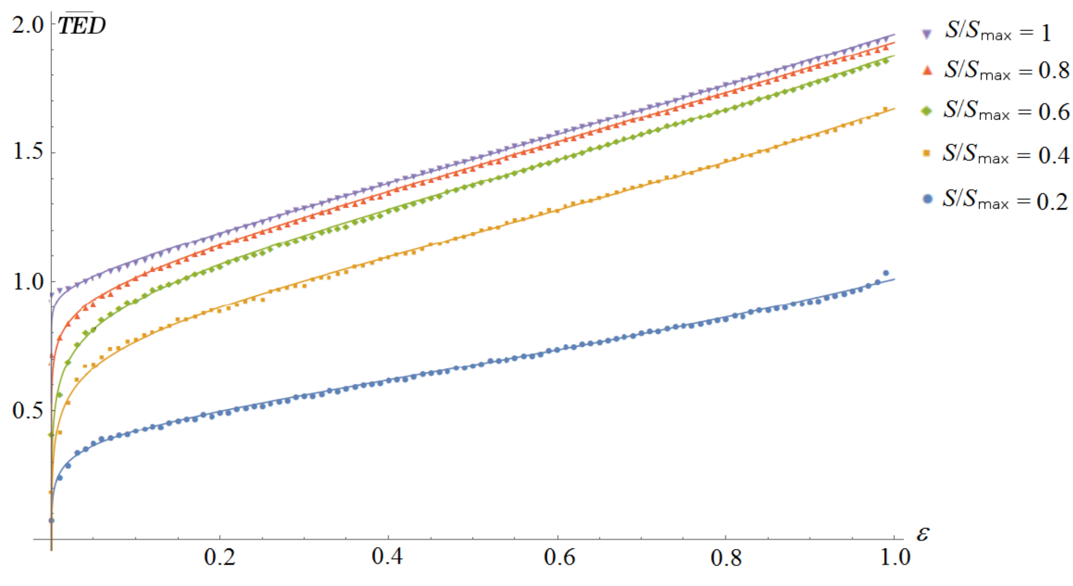

Fig. 2. Dependencies of the TED on the ε with different values of the configuration entropy S, $n = m = 48$, $l = 152$

T a b l e 1. Values of constants for dependencies of the entropy and TED

Number	1	2	3	4	5
a_0	0.631706	1.04616	1.50153	3.92962	5.29811
b_0	0.902495	1.7169	2.53405	7.66127	10.9732
c_0	0.541963	0.489177	0.392038	0.188599	0.154264
k	0.0896779	0.105287	0.0763552	0.0183298	0.00576306

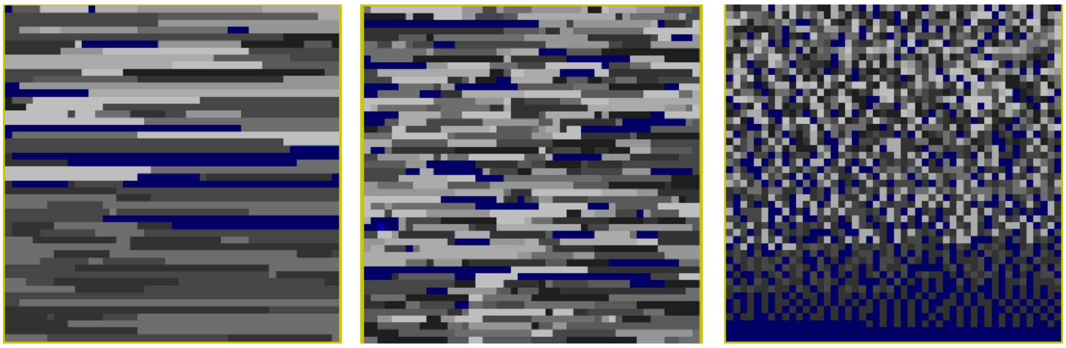

Fig. 3. Landscapes with $S = 1.90026$, $TED = 0.728733$, $\varepsilon = 0.05$, $S = 2.19722$, $TED = 1.03125$, $\varepsilon = 0.2$, $S = 1.90026$, $TED = 1.57552$, $\varepsilon = 0.9$

almost linearly. The form of such dependence only insignificantly varies with a change of the number of classes l. Therefore, it is possible to find the average value of the $TED(\mathcal{L}(N, l))$ in the form

$$\overline{TED} = a_0(S/S_{\max}) \ln \frac{c_0(S/S_{\max})\varepsilon^k}{(1 - c_0(S/S_{\max})\varepsilon)} + b_0(S/S_{\max}),$$

where a_0, b_0 are constants which depends on the configuration entropy of the landscape. Values of these constants for dependencies showed in Fig. 2, are given in Table 1.

Results of the algorithm can be seen in Fig. 3. The temporal computational complexity of this algorithm depends on the number of cells N as $O(N)$ and on the number of classes of cells l as $O(l)$.

4. The relationship between entropy and Total Edge

Suppose that the rectangular landscape $n \times m$ cells generated by the algorithm described in the section 2. The author performed 50 numerical experiments to explain the nature of the relationship between TE and configurational entropy. It became apparent during experiments that the average value of the TED for the landscape generated by the above algorithm virtually does not depend on the number of cells in the landscape, but depends only on the number of cell classes. We will search the dependence of the $TED(\mathcal{L}(n \times m, l))$ average value on l and configuration entropy S in the form

$$\overline{TED} = a_1(l)\sigma\big(b_1(l)S/\ln l\big) + a_1(l)/2,$$

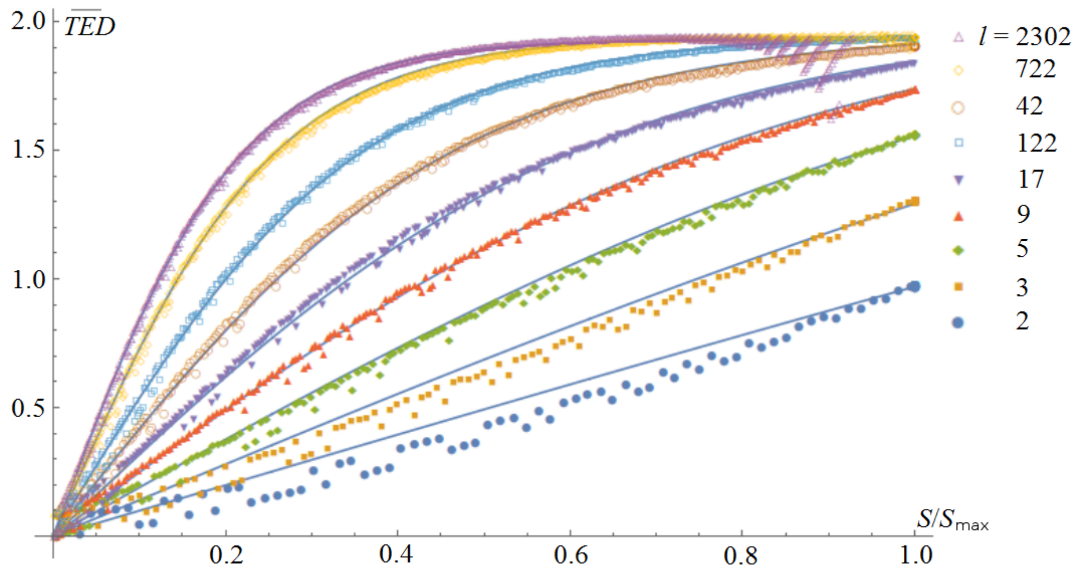

Fig. 4. The dependence of the $\overline{TED}$ on the $S/S_{\max}$ and cells classes quantities l ($n = m = 48$)

where $\sigma(x) = (1 + e^{-x})^{-1}$, a_1, b_1 are unknown functions from empirical considerations (see Fig. 4). We can obtain acceptable (with coefficients of determination $r^2 = 0.999999$ for a_1 and $r^2 = 1.0$ for b_1) approximations

$$a_1(l) = 3.90302 + 5.25218(x - 10.9641)^{-2.19235},$$

$$b_1(l) = 1.15228 \ln(0.990096x - 1.18397).$$

Conclusion

The relationship between landscape metrics was established, and the algorithm for construction of a landscape with a given Total Edge metric was developed in this article. It has been discovered that the more random the filling of a landscape with cells of different classes is, the stronger the relationship between the configuration entropy and the Total Edge of a landscape. The formula of such dependence was inferred by a computational experiment. The landscape may be filled not only from top to bottom, in a line-by-line fashion, but also by a more complex way as a further improvement of landscape generation algorithms outlined in this paper. Another possible improvement might be the creation of a landscape by some smaller landscapes with specified characteristics.

References

[1] **Rodriguez-Iturbe, I., D'Odorico Paolo, R.A.** Configuration entropy of fractal landscapes // Geophysical Research Letters. 1998. Vol. 25, No. 7. P. 1015–1018.

[2] FRAGSTATS: Spatial pattern analysis program for categorical maps. Documentation. URL: http://www.umass.edu/landeco/research/fragstats/documents/fragstats_documents.html

[3] **Kuznetsov, A.V.** A model of the joint motion of agents with a three-level hierarchy based on a cellular automaton // Computational Mathematics and Mathematical Physics. 2017. Vol. 57, No. 2. P. 340–349.

[4] **Cushman, S.A.** Calculating the configurational entropy of a landscape mosaic // Landscape Ecology. 2016. Vol. 31, No. 3. P. 481–489.

[5] **Johnson, L., Yannakakis, G.N., Togelius, J.** Cellular automata for real-time generation of infinite cave levels // Proc. of the 2010 Workshop on Procedural Content Generation in Games. PCGames '10. New York, USA: ACM, 2010. P. 10:1–10:4.

[6] A survey of procedural noise functions / A. Lagae, S. Lefebvre, R. Cook, T. DeRose, G. Drettakis, D. Ebert, J. Lewis, K. Perlin, M. Zwicker // Computer Graphics Forum 29 (2010). P. 2579–2600. URL: http://www-sop.inria.fr/reves/Basilic/2010/LLCDDELPZ10a

Received 20 December 2016

Received in revised form 7 May 2017