A Cellular Automaton for Pied-de-poule (Houndstooth)

Loe Feijs and Marina Toeters
Department of Industrial Design,
Eindhoven University of Technology,
l.m.g.feijs@tue.nl
By-wire.net, Utrecht

Abstract

We report on the generation of a specific type of pied-de-poule pattern (houndstooth) using a dedicated cellular automaton. The generator is a one-dimensional automaton such that the development features a so-called puppytooth pattern. In order to make this possible we had to introduce a system of five colors. The desired pattern appears as a contrast of darker and lighter colors, resembling the traditional black and white pattern. It is possible to extend the automaton, which offers a rich playground for various types of semi-random, yet pied-de-poule like behaviors. The pattern was used to weave fabric; we show the fabric and several realized garments.

Introduction

Pied-de-poule (houndstooth) denotes a family of patterns. The French Pied de poule means "foot of hen". We refer to Feijs' contribution to Bridges 2012 [1] and Abdalla Ahmed's work on weaving design [2] for an overview of earlier work on the computerized creation of pied-de-poule patterns. The patterns can be viewed as the outcome of a simple algorithm, but also as tessellations. The simplest pattern of the family is called puppytooth, see Figure 1. At present we are interested in cellular automata [3,4] and we want to generate fashionable patterns using simple rules, starting with puppytooth.

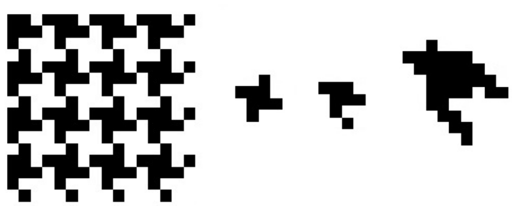

Figure 1: *Puppy-tooth pattern (left) with basic figure in windmill-Gestalt (next), basic figure in pied-de-poule interpretation (next) and example of more complicated pied-de-poule basic figure (right).*

Designing an Automaton for Puppytooth

The initial idea was to design a two-dimensional automaton such that at each point in time there is a two-dimensional grid, which resembles a pied-de-poule pattern in some areas and which evolves to locally resemble such pattern. We found rules that would sustain a given pied-de-poule pattern and we managed to add rules with a limited error-correction capability. But growing fresh pied-de-poule patterns from random seeds was harder. Then we switched to one-dimensional automata. The geometric definitions of pied-de-poule patterns are well-known [1]. We coded one such pattern in Mathematica and wrote a program to extract an automaton rules automatically. In a one-dimensional cellular automaton, the new state value is obtained by a rule from the previous-row neighbor states. A cell's environment is defined by its radius r such that r = 1 means that each environment has 3 cells. In general for $r > 1$ each environments has $2r + 1$ adjacent cells. At each point in time, $t = 1, 2, 3, \cdots$ each cell has a value (a state). The state can assume a set $\mathbf{Q}$ of k distinct values. We need a rule, which is a recipe telling how a cell is updated as a function of its environment, so for environments of three cells (r = 1 and k = 2), the rule should describe 8 cases. One such case could be (tuple) {0,0,0} → 0, which we call a *maplet*. In general, a complete rule has k^{2r+1} maplets. The automaton develops in time, and time is depicted in the vertical axis.

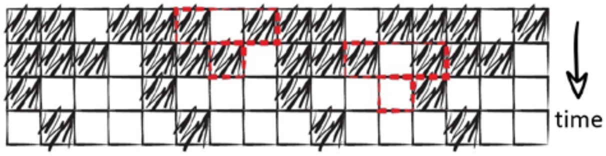

Figure 2: *Hypothesized development for t = 1,2,3,4 of puppy-tooth pattern on a one-dimensional grid of 16 cells. There is a difficulty for a formal rule to produce the pattern if we would adopt two states only.*

From Figure 2, it can be seen that no rule can perform well in making the puppytooth pattern at t = 1 and t = 2 since at t = 1 there is a need for {1,0,1} → 1 whereas at t = 2 it should be {1,0,1} → 0. Similar situations arise for {0,0,0} for example. State 0 is plotted white, 1 as black. In fact, the problem persists if the environments are chosen larger, since t = 1 and t = 2 are the same row of states, except for a horizontal shift (and similarly for t = 3 and t = 4). The problem persists for any natural number r. The proposed approach is to use k = 5 states. There is one *quiescent* state, serving as the blank space where no puppy-tooth pattern (or anything else) has developed yet. Its color is pure white. Moreover two extra kinds of white and two kinds of black are introduced in order to distinguish consecutive rows inside the puppytooth pattern at (preventing the problem of Figure 1). White and black are the colors par-excellence for pied-de-poule and thus also puppytooth in fashion. Therefore we adopt two light colors (called pinky and greeny) and two dark colors (dark-red and dark-green). The forgetful mapping F(dark-red) = F(dark-green) = black and F(pinky) = F(greeny) = white should give the classic black-and-white puppy-tooth pattern. We say pinky and dark-red are *red-like*, greeny and dark-green are *green-like*.

The coding is: quiescent = 0, pinky = −1, greeny = −2, dark-red = 1, dark-green = 2. In other words, negative values are kinds of black, strictly positive values are kinds of white. The plan is to design an automaton such that it can evolve into (regions of) puppy-tooth pattern, in which red-like and green-like rows alternate.

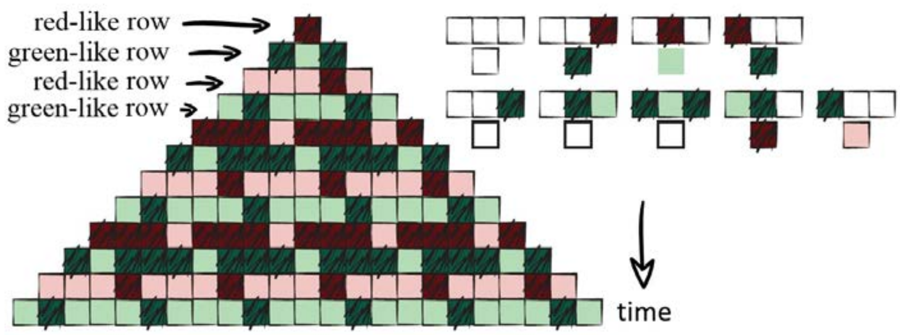

Figure 3: *First 12 rows of a puppy-tooth pattern development (left) and 9 maplets which are sufficient for development of the first two rows (t =2 and t = 3) from the initial row (t = 1).*

Figure 3 shows how a rule of 9 maplets could produce two more rows from an initial grid with a single dark-red cell. Formally we let r = 1, k = 5, **Q** = {−2, −1,0,1,2} (as a set) and then the 9 maplets are {0, 0, 0} → 0, {0, 0, 1} → 2, {0, 1, 0} → −2, {1, 0, 0} → 2, {0, 0, 2} → −1, {0, 2, −2} → −1, {2, −2,2} → −1,

{−2, 2, 0} → 1, {2, 0, 0} → −1. Continuing the development of Figure 3 we find that it takes 35 maplets to complete the emerging triangle (which begins with a single cell in state 1, a dark-red state). Of these maplets, 19 take care for growth at the edge of the blank areas (for example {0, 0, 1} → 2) and 16 other maplets sustain the development from existing pied-de-poule patterns (for example {2, −2, 2} → −1, in which no 0 occurs). As a complete rule must have $k^{2r+1}=5^3 = 125$ maplets; therefore we have considerable freedom what to do with the remaining 125−35 = 90 maplets. As a default rule we map everything else to the quiescent state, { _, _, _ } → 0, which is shorthand for the 90 maplets (everything else maps to 0). The grid is organized circularly: the boundary is wrapped around left-to-right. Instead of extending the states, it would be possible to let the state of the cell in the following generation depend on the state of its environment in this and the previous time step; but we adopted the 5-state color solution which gives certain aesthetic effects.

Implementation

The cellular automaton is programmed in Mathematica 10.4 at Eindhoven University of Technology. The fabric is woven at EE-labels in the Netherlands. EE stands for Van Engelen & Evers. The family business has been weaving quality products since 1900 in the village Heeze in North-Brabant. Van Engelen & Evers started making beautiful ribbons for saris and traditional/national costumes. Now EE supplies woven labels and other products of the very best quality to a wide range of leading global brands, see www.eelabels.com. The garments are designed and realized in the studio of by-wire.net in Utrecht.

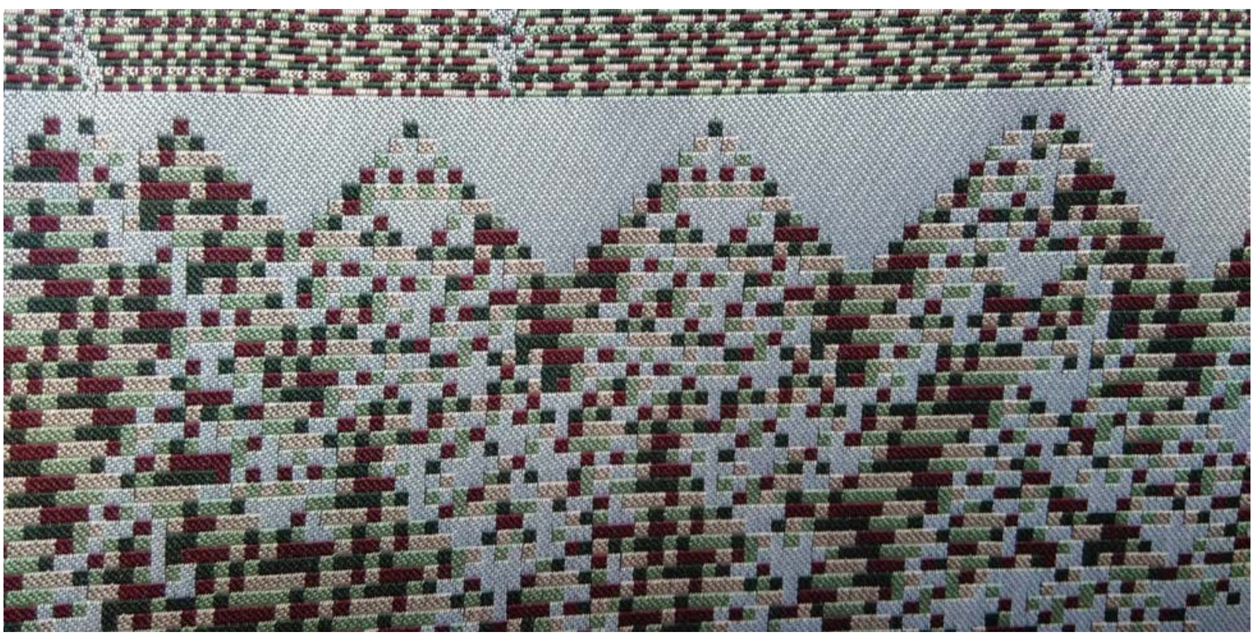

Figure 4: *Weaving the cellular-automaton generated puppy-tooth patterns with semi-random effects.*

Conclusions and Outlook

We found that it is possible to extend the automaton, which offers a rich playground for various types of semi-random, yet puppytooth-like behaviors. The semi-randomness can be seen in Figures 4 and 5. There is already emerging semi-randomness from the inherent basic puppytoooth rule. Moreover, we obtain variation in final patterns (and more semi-randomness) by the added random mutations. We believe that "emergent complexity" will be an important topic in society, in fashion and in industrial design.

Figure 5: *Two of the realized garments worn by Prof. S.M. Verduyn Lunel and Prof. J.C.M. Baeten.*

Acknowledgements: We like to thank Jun Hu, Frank Delbressine of TU/e, Robin, Joost and Frits of EE-exclusives for their expertise and kind support. We also thank Prof. S.M. Verduyn Lunel and Prof. J.C.M. Baeten for wearing the garments and the anonymous reviewers of Bridges for their helpful suggestions.

References

[1] Feijs, L.M.G. Geometry and Computation of Houndstooth (Pied-de-poule), In: Bosch. R., McKenna, D., Sarhangi, R. (Eds.), *Proceedings of Bridges 2012: Mathematics, Music, Art, Architecture, Culture* pp. 299-306. Online: archive.bridgesmathart.org/2012/bridges2012-299.pdf (access date 1-4-2017)

[2] Ahmed, A.G.M. Modular Duotone Weaving Design, In: Gary Greenfield, George Hart and Reza Sarhangi (Eds.), *Proceedings of Bridges 2014: Mathematics, Music, Art, Architecture, Culture,* pp. 27–34. Online: archive.bridgesmathart.org/2014/bridges2014-27.pdf (date of access 1-4-2017)

[3] Schiff J.L., *Cellular Automata: A Discrete View of the World*, 1st Edition, Wiley, ISBN 10:47016879X. Online: texnology.com/joel.pdf (date of access 1-4-2017)

[4] Wolfram, S. *A New Kind of Science* (NKS). Champaign, IL: Wolfram Media, Inc. ISBN 1-57955-008-8. Online: www.wolframscience.com/nksonline/toc.html (date of access 1-4-2017)