A Novel Line Fractal Pied de Poule (Houndstooth)

Loe M.G. Feijs\textsuperscript{1} and Marina Toeters\textsuperscript{2}
\textsuperscript{1}Department of Industrial Design
Eindhoven University of Technology
l.m.g.feijs@tue.nl
\textsuperscript{2}by-wire.net, Utrecht

Abstract
In earlier work we proposed a fractal pied de poule inspired by Cantor’s dust, building on a mathematical analysis of the classical pied de poule pattern. Now we propose another fractal pied de poule implemented as a single line. Instead of blocks which get fragmented into smaller and smaller blocks, we begin with a single continuous line which is expanded by adding more and more nested zigzags. Although the former approach takes a two-dimensional starting point and the latter a one-dimensional starting point, the resulting fractal dimensions are comparable (also depending on the type of the original pied de poule being fractalised). We calculate the fractal dimension and develop a fashion item based on the new pattern, to be shown at Bridges.

Introduction
After our earlier mathematical analysis of the classical pied de poule pattern [1] and the invention of a fractal version of it [2] we felt that there was not much more to be explored in houndstooth. Then in 2014 we presented an innovative line fractal [3] (not pied de poule) for which we found turtle graphics [4] to be a useful tool. In contemporary fashion, pied de poule is still very much alive. It is an important cultural and mathematical phenomenon [5, 6]. So at some point we asked ourselves whether it would be possible to construct a line fractal for pied de poule.

![Figure 1](image)

**Figure 1:** Christian Dior’s 2012 design(left), detail of the same, chain of basic pied de poule figures of type two and type one (with zigzags) and with recursive zigzags (rightmost figure).

We got inspiration from one of Dior’s designs of 2012, Fig. 1 (a) which we also mentioned in [1] (mainly noting that the basic figures are isolated and appear to fly out). But zooming in on the same design, as in the second picture of Fig. 1, we now noted something else: that each basic figure was in fact a kind of zigzag
line. We already knew that inside a classic pied de poule pattern, the black basic figures are connected and thus form chains, as shown in Fig. 1 (third picture). Zooming out, such a chain could be considered a kind of line. Perhaps the zigzags could be chained and at the same time the line drawing could be done zigzag-wise in a recursive manner (last two pictures).

From [1] we recall that there is not just one pied de poule, but an indexed family, one pied de poule pattern for each integer $N > 0$. This is shown in Fig. 2. We deployed three types of mathematical tools for our analysis: (1) compact formulas to describe the twill binding and the color patterns of the warp and the weft, (2) Heesch and Kienzle’s theory to describe how the basic figures are tessellated, and (3) turtle graphics to describe the contours. In Fig. 2 we show the pied de poule patterns for $N = 1, 2, 3$, and 4. Therefore we decided that it would be better not to look for just a single fractal but rather for a family, one for each $N$.

![Figure 2: Successive pied de poule patterns for N=1,2,3,4 from [1].](image)

Since we aim at a line fractal design, we also have to mention some of the known examples of line fractals, see [7], such as the Koch curve, the dragon curve, the Sierpiński arrowhead curve, the Hilbert curve, Kaplan’s smooth self-similar curves [8] and our own warp knitting fractal [3].

**Requirements**

So we set out to search for a fractal line which would embody some of the key principles from [1] and would be recognisable as a pied de poule pattern. The initial explorations were promising, as demonstrated in Fig. 1. But except for the case $N = 1$ the explorations appeared difficult and messy as we found several fractal lines which were pied de poule-like, but had a deficiency of one kind or another. We had to structure the process first, defining more precisely what would be a good fractal line pied de poule. Since there is not a single classical piede poule, we aimed at a family of fractal lines, not just a single solution. In summary, our requirements are that it has to be pied de poule-like, recursively tessellated, parameterized, generic and continuous. We explain each of the requirements in detail now:

- **Pied de poule-like:** for each $N$ the line should resemble the classic basic pied de poule as formalised in [1], which in turn originates in the weave and in the block patterns of the warp and the weft (“$N$-over, $N$-under” weave and patterns of $2N$ white threads followed by $2N$ black threads).
- **Recursively tessellated:** the number $n$ is the recursion level such that the figure for level $n$ should look like a tessellation of figures for type $(n-1)$. It should be tessellated in the characteristic pied de poule style, which implies that the empty space between the figures has the same form as the figures themselves (although it would be okay for this same-ness only to appear as a limit case).
- **Parameterized:** we demand a family of lines, one for each $N$ (which can be approximated for each $n$).
- **Generic:** the figures should be described or generated by a generic recipe with a minimum of ad-hoc tricks and which works the same for each $N$ and each $n$.
- **Continuous:** each line is a continuous line without jumps.
Initial explorations

We found a line fractal based on a zigzag inscribed in the basic figure of the classic pied de poule for \( N = 1 \): the long diagonal lines are implemented as a sequence of recursive zigzags, whereas the short lines, which serve for shifting to the next diagonal, will be done just straight, as indicated in Fig. 1 (right). The recursive zigzags of adjacent lines can be aligned such that they form a pied de poule-style tessellation. We implemented this using turtle graphics (which also has been an important tool in our earlier works such as [1, 2, 3, 4]). The turtle graphics command sequence for the downgoing diagonals has to be reversed with respect to the command sequence for the upgoing diagonals. In Fig. 1 (right) the turtle begins at the lower end of the entire chain.

The fractal line pied de poule of Fig. 1 (right) was quite satisfying and we set out enthusiastically to generalise this to larger \( N \) values. This turned out difficult. To illustrate the nature of the difficulties we refer to Fig. 3. From the classic figure taken from [1] we see that the number of diagonals is always even. But it seems we have to begin with an up-going diagonal and we have to end with an up-going diagonal, which suggests an odd number diagonals. When we take shortcuts to make ends meet, we are always left with a gap inside the figure. Additional deficiencies arise with respect to the tessellation, where the main figure and the complement figure either overlap or leave gaps too.

Even if we gave up on the idea that the diagonals had to be well-aligned with respect to the original grid of the classic pied de poule, we found no solutions which satisfied all the requirements of the section “requirements”. We did not want to let the turtle walk back over its own trace (which would have been an ugly solution with lots of ambiguity). But at some point we took another fresh look at the concept of continuity and its opposite, discontinuity, which in turtle graphics means “pen up” and “pen down”: the solution presented in the next section.

The novel fractal line pied de poule

Now consider the following idea: if we would be allowed to use pen-up and pen-down turtle graphics commands, then we could draw all the essential diagonals and connect them by special line segments and arcs. The special line segments and arcs would be outside of the classic figure, but we could draw them with pen-up and thus they would not be harmful. Or perhaps we could draw them with a very thin pen and they would be “almost” not harmful. This is shown in Figure 4.
Instead of having a pen-down command, we let the drawing function work recursively, writing pied de poules all along. In order to make sure the figures tessellate correctly, we have to do two diagonal pied de poule figures in each cell. So they shrink by a factor of $\frac{1}{8}\sqrt{2}$ (for $N = 1$), by $\frac{1}{16}\sqrt{2}$ (for $N = 2$) and by $\frac{1}{24}\sqrt{2}$ (for $N = 3$). In general they shrink by a factor $\frac{1}{8N}\sqrt{2}$. The effect is demonstrated in Figure 5 for $N = 3$.

Figure 4: Drawing the diagonals of a classic pied de poule with outer loops drawn with a thinner pen for $N = 1$ (left), $N = 2$ (center) and $N = 3$ (right).

Figure 5: Fractal line pied de poule approximation: solution for $N = 3$ and $n = 2$ based on the infinitesimally thin outer-loops concept. © Loe Feijs.
The outer loops are needed to connect the zigzagged diagonals. Each outer loop consists of a line segment and an arc, which is a half circle. The choice to use half circles is made for aesthetic reasons (we tried straight lines but that appeared ugly). The half circles make the construction less edgy and are neutral with respect to directionality. We can only draw approximations, such as for example the approximation corresponding to $n = 2$ in Fig. 5. The effect is that the supposedly “thin” lines are thin indeed. We call them *infinitesimally* thin lines. In our practical approximations, they are visible (but thin indeed). The whole figure consists of a single line which does not cross itself. This solution works for all $N$. It is a remarkable feature of this construction that the figure rotates by 45° at level $n - 1$, by 90° at $n - 2$ and so on. The line touches at certain points, but we can tweak this almost invisibly and have one long non-intersecting line.

Intuitively we can say that the outer loops are a minor thing, but can we prove it in a formal sense? It turns out that adding an $\varepsilon$-fattening band around the classic pied de poule is enough to let it cover the fractal line pied de poule approximation, including the protruding outer loops. We find that this $\varepsilon$ vanishes for $N \to \infty$. So we can neglect the outer loops for large $N$, but even for fixed $N$ we can show that the practical drawing of the outer loops must be of a thickness going to zero when approximating for $n \to \infty$.

The formulation of the formal properties and the mathematical proofs are outside the scope of this paper.

**Fractal dimension**

The fractal dimensions are comparable to those of the earlier Cantor-dust inspired fractals. For $N = 1$ we refer to Fig. 4 (left). Replacing a line segment of length 1 by a zigzagged pied de poule we replace it by 16 ‘line’ segments of length $\frac{1}{8}\sqrt{2}$ each. Writing $m$ for the number of line segments, $s$ for the scaling factor, $m = 16$ and $\frac{1}{s} = 1/(\frac{1}{8}\sqrt{2}) = 4\sqrt{2}$ so the dimension $D(1) = (\log m)/(\log \frac{1}{s}) = (\log 16)/(\log 4\sqrt{2}) = 4/2 = 1.6$. For $N = 2$ we find $m = 64$ and $\frac{1}{s} = 1/(\frac{1}{16}\sqrt{2}) = 8\sqrt{2}$ so $D(2) = (\log 64)/(\log 8\sqrt{2}) = 6/3 = 2 = 1.7143$. For $N = 3$ we find $m = 144$ and $\frac{1}{s} = 1/(\frac{1}{24}\sqrt{2}) = 12\sqrt{2}$ so $D(3) = (\log 144)/(\log 12\sqrt{2}) = 1.7552$.

In general we find $m = (4 \times N)^2$ and $\frac{1}{s} = 1/(\frac{1}{8N}\sqrt{2}) = 4N\sqrt{2}$ so $D(N) = (2\log 4N)/(\log 4N\sqrt{2})$. Compare these results with the earlier fractal pied de poule (called fPDP) [2]: $D_{fPDP}(1) = 1.5$, $D_{fPDP}(2) = 1.6667$, and $D_{fPDP}(3) = 1.7211$. Again $\lim_{N \to \infty} D(N) = 2$ but convergence is slow as for $D(N) = 1.99$ we need $N = 10^{30}$.

**Application**

Once we had found a satisfying fractal line pied de poule we considered the medium to apply it: embroidery and/or laser engraving (see Fig. 6).

![Figure 6: Embroidered sample (left) and laser-engraved sample (right).](image)
Modern embroidery machines can read the stitches from a file. We programmed a layer of Processing (Java) software around the open-source turtle graphics library Oogway [4], which we named Stitchway. It supports turtle graphics like Oogway and besides generates a file in the ternary Tajima file format, which can be interpreted by Brother embroidery machines and embroidery simulators such as TrueSizer by Wilcom. Then we made both embroidered and laser-cut samples. A first sample of embroidered textile is in Fig. 6 (left) and another sample engraved in multi-layer woven textile is in Fig. 6 (right).

We have designed and constructed a mini-collection of three attractive high-tech fashionable garments based on the new fractal line pied de poule:

- a body stocking,
- a parka (coat), and
- a jacket.

The body stocking and the parka can be seen in Fig. 7 (left) whereas the body stocking and the jacket can be seen in Fig. 7 (right). A larger photograph of the parka is in Fig. 8.

![Figure 7: Fractal pied de poule collection 2015: Body stocking and Parka (left), Body stocking and jacket (right). Lasercut multilayer woven polyester fabric, model Renata van Putten, photo Brian Smeulders, © Marina Toeters.](image-url)
Conclusions

The search for this fractal was quite an adventure: the attempts illustrated in Fig. 3 are just a few from an almost endless series of failing attempts to solve the puzzle (for which we did not know beforehand whether there was a solution waiting for us). We consider the zigzags with infinitesimally thin outer loops a very elegant solution. It allows us to completely avoid the ad-hoc clumsiness of pen-up/down commands or line thicknesses. Yet in the theoretical sense of $N \to \infty$, the zigzags do not protrude out of the classic pied de poule figure. Moreover, in another theoretical sense, viz. approximating for $n \to \infty$, the outer loops are invisible anyhow. The solution has no further ambiguities: if we formally define a zigzag to be any alternation of straight lines and arcs (semicircles) without sharp bends, then there is a unique zigzag of minimal length which covers the diagonals of the classic pied de poule beginning and ending at the connector points of the classic chain. Inside the classic figure the pen is down, meaning there is recursion; outside the pen is up, that is: no recursion.

Practically, this solution allows us to choose any continuous medium such as stitching, embroidery, plotting (without lifting the pen), bended wire, bended pipes and so on. Using wire, conductive yarn, or conductive ink, the figure turns into a resistor, or an antenna, or even a sensor for touch\(^1\). The novel line fractal pied de poule yields figures with dimensions which are very close to those of our earlier fractal [2], even though the former begins from a one-dimensional figure (the line) which is extended by zigzagging whereas the latter begins as a two-dimensional figure (a classic pied de poule) which is Cantor-dustified by cutting out sub-squares. But the fractals are not the same, for example the 45° rotation is a unique feature of the novel construction. Finally, we thank Jun Hu, Lilian Admiraal, Chet Bangaru, Jasper Sterk and Jan Rouvroye for their kind support.

References


---

\(^1\)In fact our long term ambition is to combine the experience we have gathered in this project with other sensor and actuator projects into a more complex and interactive system (our ambition for next year’s Bridges).
Figure 8: Fractal pied de poule collection 2015: Parka. Lasercut multilayer woven polyester fabric, model Renata van Putten, photo Brian Smeulders, © Marina Toeters.
Design of a Nature-like Fractal Celebrating Warp-knitting

Loe M.G. Feijs1, Marina Toeters2, Jun Hu1, and Jihong Liu3
1Department of Industrial Design
Eindhoven University of Technology
l.m.g.feijs@tue.nl
2by-wire.net, Utrecht, 3Jiangnan University,

Abstract
In earlier work we created a new textile pattern which was derived from the well-known houndstooth pattern which originates from weaving with twill binding. The new pattern became interesting, both mathematically and aesthetically because it was a fractal. Now we are turning our attention to another basic fabric construction method: warp-knitting. We develop a recursive algorithm and explore the properties of the result. We also develop an attractive fashion item based on the new pattern, to be presented at Bridges.

Introduction
First we explain what warp-knitting is and in which sense we take inspiration from it. The typical characteristic of knitting is that the threads form loops, each loop being pulled through an existing loop. Roughly speaking, there are two main approaches to knitting, called weft-knitting and warp-knitting. The well-known hand-knitting is a special case of weft-knitting, for example a single yarn being knitted from left to right and then from right to left. In warp-knitting however, the yarn moves in the length-direction of the fabric in a zigzag manner [6]. So, unlike weft-knitting, a warp knitted fabric is composed of many yarns, not just one.

Figure 1: Warp knitted fabric (left) and one thread thereof (right).

Earlier work by Bernasconi, Bodie and Pagli on algorithmic knitting [1] demonstrates the power of recursion as a programming technique for knitted fractals (we use recursion as an essential tool too). The work of the present paper is the result of a new cooperation between TU/e, Jiangnan University and by-wire.net which was initiated during the DeSForM2013 conference in Wuxi. Whereas the Industrial Design Department in Eindhoven has strength in wearable senses and in generative design, the Engineering Research Center of Knitting Technology at Jiangnan University, Wuxi is specialised in warp-knitting. We share an interest in textile design and algorithmic pattern design, witnessed by results such as [4, 3, 2].

In Figure 1 (source of left figure: Wikimedia Commons) the basic principle of warp-knitting is given. One yarn is singled out and this one yarn with its loops is taken as the inspirational source for the fractal to be designed. But first we explain a bit of fractal theory.
How to make a fractal

We take inspiration from line fractals such as the Koch fractal and the dragon curve. Lindenmayer systems [5] are often used to describe the growth of fractal plants. This works with substitution, e.g. a forward move $F$ can be replaced by $F$→$F$+$F$+$F$+$F$. As a formal rule: $F$→$F$+$F$+$F$+$F$+$F$+$F$. The idea is to apply the rule repeatedly (to all $F$ simultaneously). Starting from $F$ we get $F$+$F$+$F$+$F$, then $F$+$F$+$F$+$F$+$F$+$F$+$F$+$F$+$F$+$F$+$F$+$F$, and so on. Interpreting the symbols as turtle graphics commands, one gives $F$ the meaning of drawing forward, + to turn right 60°, and − to turn left 60° and then this Lindenmayer system describes the Koch fractal.

The warp-knitting fractal

We show the approximations of our new fractal for nesting levels $N=0,1,2,3$ and 4 in Figure 2. These have been created using a recursive algorithm and a turtle-graphic system, in a similar way in which one makes the Koch fractal. The lines in Figure 2 are drawn starting at the bottom of the figure with the turtle pointing upward. For the second line of Figure 2, the turtle made two loop pairs. In this way we get loops similar to the single yarn of Figure 1 (the loops are not nicely rounded yet, but we will repair that later).

This gives us a recipe for a fractal: draw a looped line, but whenever the basic recipe tells us to move forward, we move forward while doing a few loop pairs. More precisely: we do 3 loop pairs for the first “forward”, 2 for the next (it is shorter by a factor of $2 \sin 15^\circ$), then 3 again, and 4 for the last “forward”. And then we repeat in a glide-mirrored fashion. The numbers are chosen after experimentation: 2 for the shortest line, 3 inside the loops (where the corners would become messy otherwise) and 4 for the last move. The recipe is related to the Lindenmayer rule $F$→$-F^3-F^2-F^4+F^3+F^2+F^3+F^4$ where $F^2$ abbreviates FF, $F^3$ abbreviates FFF and so on and where the four minus signs represent left turns of 30°, 105°, 105° and 90° respectively; the plus signs represent right turns of 105°, 105°, 90° and 30° (to specify the exact lengths we would need the more powerful formalism of parametric L-systems). In practice we use the Oogway library in Processing [2]. This also allows us to fine-tune the scaling factors of subfigures and explore aesthetic effects.

Fractal dimension

Replacing a line of length 1 by a loop pair, it turns into six segments of length $\frac{1}{3} \sqrt{3} \approx 0.577$ and two of length $\frac{1}{2} \sqrt{3} \sin 15^\circ \approx 0.299$. If we replace it by three loop pairs, it turns into 18 segments of length $\approx 0.577/3 \approx 0.19$ and 6 of length $\approx 0.299/3 \approx 0.10$. So one line is replaced by 24 segments of a (weighted) average of length 0.17. In the fractal, most lines are replaced by three loop pairs, but there are

Figure 2: Approximations of the warp-knitting fractal for $N = 0, 1, 2, 3$ and 4.
also those which are replaced by two loop pairs or by four. To estimate the dimension we pretend each line is replaced by three double loops, so it is broken up in 24 segments of length $\approx 0.17$. Writing $n$ for the number of line segments, $s$ for the scaling factor, $n = 24$ and $\frac{1}{s} = 1/0.17 = 5.9$ so $D \approx (\log 24)/(\log 5.9) = 1.8$.

The fractal is almost two-dimensional, which is what we see in the rightmost line of Figure 2: the line almost appears to fill certain areas. This gives the line its natural appearance, like a plant. If we insist on avoiding approximations, we solve $n_1 \times s_1^D + n_2 \times s_2^D + n_3 \times s_3^D = 1$ where $n_1 = 12$, $s_1 = (\frac{1}{3} \sqrt{3})/3$, $n_2 = 8$, $s_2 = (\frac{1}{3} \sqrt{3})/4$, $n_3 = 4$, and $s_3 = (\frac{2}{3} \sqrt{3} \sin 15^\circ)/2$. Using Mathematica’s FindRoot we get $D = 1.79659$.

Back to fashion

We promised to make rounded loops, which we achieve using `beginSpline` and `endSpline` in Oogway [2]. This strengthens the nature-like appearance and even for low $N$ it resembles a vine plant now (Figure 3, left). The next step is designing a real fashion item: an elegant lady’s dress. We used a combination of knitting (the jersey substrate) and textile printing (the fractal line); special thanks go to Pauline Klein Paste of HKU (Utrecht School of Arts). The pattern can be seen in Figure 3 (center) and the dress in Figure 3 (right) and Figure 4. An interesting question is whether the new pattern can be really machine-knitted. It will also be interesting to see what happens if we involve multiple threads (we leave these questions as options for future research). We shall bring the dress to Bridges Seoul.

![Figure 3: Spline-based line (left), pattern (center) and lady’s dress with pattern of line fractal (right), (Model Charlotte Geeraerts, Make-up artist Lana Houthuijzen, Photographer Katinka Feijs).](image)

References

Figure 4: Lady's dress with green fractal (Model Charlotte Geeraerts, Make-up artist Lana Houthuijzen, Photographer Katinka Feijs).
Constructing and Applying the Fractal Pied de Poule (Houndstooth)

Loe M.G. Feijs¹ and Marina Toeters²
¹Department of Industrial Design
Eindhoven University of Technology
l.m.g.feijs@tue.nl
²by-wire.net, Utrecht

Abstract

Time is ready for a fractal version of pied de poule; it is almost “in the air”. Taking inspiration from the Cantor set, and using the analysis of the classical pattern, we obtain a family of elegant new fractal Pied de Poules. We calculate the fractal dimension and develop an attractive fashion item based on the new pattern, to be showed at Bridges.

1 Introduction

In earlier work we analyzed the mathematics behind the classical Pied de Poule, also called Houndstooth (see Fig. 1) with tools such as tessellation theory, compact Processing programs and compass Logo [1]. In this section we shall argue that time is ready for fractal pied de poule. The fashion community is awaiting and subtly announcing this innovation, yet not quite sure how to do it properly.

In Fig. 1 (b) we see Alexander McQueen’s 2009 ladies bag with mixed large and small PDPs (pied de poules). Figure 2, (a) is one of Dior’s designs of 2012 (there exists a similar design by Neil Barrett for men 2009). The PDP figures are isolated and appear to fly out. In December 2012 we spotted the jacket by Gerry Weber (Fig. 2, b) with almost-PDP figures inside the main-level figures (yet the knitwork has not a precise classical PDP). Even PDP antagonist blogger Anti-Houndstooth saw the possibility coming in 2009: “Sometimes I wake up in a cold sweat to the idea that one day some slick mathematician will discover the fabric of the universe is a houndstooth weave and the mandelbrot set will reveal universe after universe of pulsing houndstooth patterns!!” [4]. In 2012 we began searching for fractal pied de poule (fPDP), main results being communicated at the mini-symposium “mathematics and art” of Eurandom, Eindhoven, July 5, 2012 and published now for the first time.

2 Towards a fractal

We take inspiration from the well-known Cantor set. This is a subset of the unit interval \( \{ x \in \mathbb{R} \mid 0 \leq x \leq 1 \} \). The subset is formed by splitting the interval in three segments and removing the middle part \( \{ x \in \mathbb{R} \mid \frac{1}{3} < x < \frac{2}{3} \} \). This process can be repeated on the two remaining parts, and so on. The set of points not removed is

Figure 1: Successive PDP patterns for \( N=1,2,3,4 \) from [1] (a) and McQueen’s bag (b).

Figure 2: Christian Dior’s 2012 (a) and almost fractal PDP by Gerry Weber 2012 (b).
the Cantor set. It is a fractal: it is equal to two copies of itself, each shrunk by a factor of 3 and translated [3].

![Figure 3: Approximating Cantor set (a), Cantor dust (b), and twill-woven Cantor-set warp and weft (c).]

The first idea was to replace the black-white pattern of the weft and of the warp by a Cantor set each. Weaving in twill binding we hoped for a fractal PDP. Regretfully this did not work out, we got sparse grids which looked neither fashionable nor PDP-like (Figure 3 c). Then we turned to re-using the results of [1]:

- for each \( N > 0 \) there is a PDP pattern which can be compactly described by two nested for loops with loop-counters \( i \) and \( j \) and a compact Boolean formula in \( i \) and \( j \).
- for each \( N > 0 \) there is a unique figure consisting of \( 8N^2 \) squares such that an equal number of black and white tiles fit together in a tessellation which is precisely the classical PDP pattern of type \( N \).

An example Boolean formula is \( (i - j) \% 4 < 2 ? i \% 8 < 4 : j \% 8 < 4 \), which produces the second pattern from Fig. 1 \((N = 2)\). In Fig. 4 we show the two basic figures for \( N = 1 \) and \( N = 2 \), alongside two "faux" figures which do tile to a PDP, but are not basic since they do not correspond to the most symmetric and compact compass Logo contour description. The basic figures converge towards the limit figure.

![Figure 4: PDP figures for \( N = 1 \) (a,b), tiling "faux" figures (c,d), figure b in grid (e), and limit figure (f).]

The new recipe is: take one figure and replace each of its \( 8N^2 \) squares by a scaled down figure, surrounded by an equal amount of white. For scale-down factor \( 4N \) an elegant fractal pattern arises which visually relates to PDP, and which we call fPDP (fractal PDP). E.g. if \( N = 2 \), let \( e(i, j) \) for \(-2 \leq i, j < 10\) be the Boolean function defined by Figure 4 (e), where black means true. Consider the recursive program:

```c
void fPDP(float d, float x, float y, float S){
    if (d <= 0)
        rect(x,y,S,S);
    else for (int i = -N; i < 5 * N; i++)
        for (int j = -N; j < 5 * N; j++)
            if (e(i,j))
                fPDP(d - 1, x + i*S / (4*N), y + j*S / (4*N), S / (4*N));
}
```

The \( d \) regulates recursion depth and \( \text{rect}(x,y,w,h) \) draws a black rectangle of width \( w \) and height \( h \). Intuitively the fractal is \( \lim_{d \to \infty} \mathcal{B}(fPDP(d,x,y,S)) \) where \( \mathcal{B}(f) \) is the closed subset of \( \mathbb{R}^2 \) marked black by \( f \). In the notation of [5] (Hutchinson’s iterated function system) we have a family of \( 8N^2 \) contractions \( S_{i,j} : \mathbb{R}^2 \to \mathbb{R}^2 \) so that \( |S_{i,j}(x) - S_{i,j}(y)| \leq c|x-y| \) with contractivity factor \( 0 < c < 1 \) viz. \( c = 1/4N \) and hence

\[ D = \log 2 / \log 3 = 0.6309 \]

\[ D = \log 4 / \log 3 = 1.2619 \]

\[ D = \log 2 / \log 3 = 0.6309 \]

\[ D = \log 4 / \log 3 = 1.2619 \]
there exists a compact non-empty set $F$ satisfying:

$$F = \bigcup_{-N \leq i, j < N} S_{i,j}(F)$$

In practice we work with finite $d$. The fractal approximated by Figure 5 for $N = 1$ is uniform in the sense that the subfigures, sub-subfigures etc. are all of the same $N$-type. We can make patterns with the $N$-type increasing for sub-figures, or even differ per position. The results with decreasing $N$ are practical, but stopping at or before $N = 1$, formally not fractals.

**Figure 5:** Fractal Pied de Poule (fPDP) approximations for $N = 1$ ($d = 2, d = 3$ and $d = 4$ respectively).

### 3 Application

The next step is designing a real fashion item, an elegant men’s shirt. The result is shown in Fig. 6. This is a non-uniform fPDP where the main figure has $N$-type 3, the subfigures have $N$-type 2, the sub-subfigures have $N$-type 1 and then the recursion stops. The pattern is generated by the recursive algorithm in Processing, post-processed in Adobe Illustrator and cut with a Speedy300 laser cutter at TU/e. The shirt was designed and welded in the fashion technology studio by-wire.net. The outer layer of white fabric has been laser-cut and the tiny holes reveal the black layer underneath. We will bring the shirt to Bridges and show it in action.

### 4 Fractal dimension

For uniform fPDP we calculate the fractal dimension using box-counting [5]. E.g. for $N = 2$, if at some zoom-in depth, one figure can be covered by a disc of diameter $\epsilon = \epsilon_0$ then at the next level it has 32 sub-figures, each covered by a disc of diameter $\epsilon = \epsilon_0/8$. The next level needs $n = 32^2$ discs of diameter $\epsilon = \epsilon_0/64$. At depth $d$ it takes $n = n(\epsilon) = 32^d$ discs of diameter $\epsilon = \epsilon_0 8^{-d}$ to cover the fPDP. The fractal dimension is defined as

$$D = \lim_{\epsilon \to 0} \frac{\log n(\epsilon)}{\log 1/\epsilon}$$

so $D_2 = \lim_{\epsilon \to 0} \frac{32^d}{\log(1/\epsilon_0 8^{-d})} = \lim_{8^{-d} \to 0} \frac{\log 32^d}{\log 8^d - \log \epsilon_0} = \lim_{8^{-d} \to 0} \frac{\log 32^d}{\log 8^d - \log \epsilon_0} = \frac{\log 32}{\log 8} = 1.5849$. For $N = 1$ we find $D_1 = 1.5$, for $N = 3$, $D_3 = 1.7211$. Generally $1\frac{1}{2} \leq D_N < 2$ for $1 \leq N < \infty$ and $\lim_{N \to \infty} D_N = 2$ (i.e. Fig. 4, f, viewed as fractal has infinitesimally shrunk white space inside).

### 5 Conclusions

It was in the air, now fractal Pied de Poule is a reality. The proposed patterns are new to the best of our knowledge. The work of Fig. 6 will also be shown at the Bridges art exhibition. We contacted the author of Anti-houndstooth [4] and to our happy surprise he was very positive: “I had long given up hope that there was any chance of redemption for the abused and maligned pattern. I am happy to find that your project
reinvigorates the iconic motif with energy and purpose.” An interesting question is whether the new pattern can still be woven on a traditional loom (we leave that as an option for future research). We thank Chet Bangaru and Jasper Sterk for their support.

References

Geometry and Computation of Houndstooth (Pied-de-poule)

Loe M.G. Feijs
Department of Industrial Design
Eindhoven University of Technology
l.m.g.feijs@tue.nl

Abstract
We apply a variety of geometric and computational tools to improve our understanding of the Houndstooth (Pied-de-poule) pattern. Although the pattern must have been known for centuries, it was made famous mostly by Christian Dior and is still frequently used in many variations. It is a non-exhaustible source of inspiration for fashion designers.

1 Introduction
Houndstooth denotes a family of patterns. The French Pied de poule means “foot of hen”. We create computer programs generating the patterns, trying to capture the essence of Pied de poule in programs which are as compact as possible. We use these as tools to see what happens in the extreme cases of the coarsest and finest possible grids. We find for example that, in a precise sense, the English term “puppy-tooth” is more correct than the Dutch “molentjes” for the simplest possible pattern. We find, as expected, that all patterns can be described by the Heesch-Kienzle theory of tilings (as are many works by M.C. Escher). Somewhat unexpectedly, we find that the limit case has different Heesch-Kienzle type than the regular patterns.

This section introduces the origin and applications of the pattern. In Section 2 we create compact programs generating the pattern (one for each \( N > 0 \)). In Section 2 we let \( N \) go to infinity. Tesselations theory is used in Section 3. The contours of the basic figure are tackled using special turtle commands (Section 4). Section 4 gives a fresh look at the very smallest figure. Feeding the computer-generated contours into another modern tool, a laser cutter, was also a source of fresh ideas: see Section 5 for novel patterns.

Weaving is the process of creating a fabric from 2 perpendicular thread sets. One set is called the warp (vertical), the other the weft (horizontal). The weft thread crosses either over or underneath the warp threads and the precise about rule where to cross over and where to cross underneath determines the weave type of the fabric thus made. Examples are plain, twill and satin weave. In twill weaving the pattern of the warp going up and down is more involved than just one-up one-down. A frequently used twill is the 2/2 twill (two warp up and two warp down). For each subsequent weft thread, the pattern is shifted by one position. For a longer explanation see for example [1] or [2]. Twill is the preferred weave for kilts, overcoats, uniforms etc.

Let the warp threads be colored according to a simple one dimensional pattern, e.g. 4 white, 4 black, 4 white, 4 black and so on. In the same way the four white, four black repeated pattern can be used for the weft threads. In combination with the a 2/2 twill weaving, the most interesting pattern of Figure 1 appears. It is known by the French term Pied de poule, in English it is Houndstooth and in Dutch Hanevoet. Whereas printed patterns (flowers, Paisley print) are vulnerable to wear and becoming faint after washing, the woven pattern is more robust: the threads were dyed before the weaving (and printing is often one-sided). The woven pattern is sharp: the edges of the pattern coincide with transitions between threads. We conjecture this to be one reason why the Pied de poule pattern is often taken as a symbol of high quality fashion (next to the robust nature of the fabric, cf. kilts, overcoats and uniforms). Later, printed Pied de poule was produced. As shown in Figure 1 (right) it is applied in contemporary fashion (e.g. in the 2009 winter collection of the Preen label). Other labels like Moschino, DKNY, Stella McCartney, and Alexander McQueen also use it.
The warp-up warp-down effects can be simulated by a digital computer program. Our first programs mimicked the behavior of the loom, describing the construction of weft threads, going from left to right. This works (Fig. 2 left) but is impractical as the algorithm must be reworked for other patterns. Next we made table-driven programs. We provide them via the Bridges CD-rom and www.idemployee.id.tue.nl/l.m.g.feijspieddepoule.html. Two more screen shots of such program’s output are in Fig. 2.

A table-driven program uses modular arithmetic for array indices. The warp and the weft colours are stored in 1-dimensional arrays (1 is white, 0 is black). The weaving pattern is in a $4 \times 4$ array (or $8 \times 8$, the program is easily adapted). The value is 1 or 0 depending on which of warp and weft is on top. We can experiment with the patterns.

Consider $W_1 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$ and let $W_2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

The middle pattern in Fig. 2 is from $\begin{align*} \text{warp} &= \{1,1,1,1,0,0,0,0\} \\ \text{weft} &= \{1,1,1,1,0,0,0,0\} \end{align*}$ and $\text{weave} = W_1$. It is another checkerboard, seen in cook’s clothing. The right hand side pattern in Figure 2 is from $\begin{align*} \text{warp} &= \{1,0,1,0,1,0,1,0\} \\ \text{weft} &= \{1,0,1,0,1,0,1,0\} \end{align*}$ $\text{weave} = W_2$. It is a kind of herringbone (Dutch: visgraat).

2 **Compact characterizations**

Then we continued searching for even more compact characterizations. There is something very specific about the well-known Pied-de-poule, herringbone or checkerboard patterns. The warp and weft arrays are
filled with simple alternations like 1 with 0 or 1,1 with 0,0 or 1,1,1,1 with 0,0,0,0. The notion of complexity
as proposed by Kolmogorov is a source of inspiration. According to Kolmogorov [3], the entropy of an x
is is the length of a shortest program \( P \), itself written as as sequence of zeroes and ones, that outputs \( x \). We
should characterize our patterns by formulas that are as compact as possible.

Let \( M \equiv m \) be the integer number \( x \) such that \( 0 \leq x < m \) and \( x \equiv M(\text{mod } m) \). We define a Boolean
function of coordinates, \( i \) and \( j \), say, such that the outcome \( \text{true} \) means that the area designated by \( i \) and \( j \)
becomes white; \( false \) means black. The notation \( \omega \) is if-then-else. Here is such a compact formula:

\[
a_{ij} = (i-j)\%6 < 3 \ ? \ i\%12 < 6 \ : \ j\%12 < 6
\]

The condition \( (i-j)\%6 < 3 \) determines whether the horizontal (weft) thread is chosen or the vertical thread
(the warp). In the former case \( i \) determines the color, in the latter case \( j \) determines the color. The condition
\( (i-j)\%6 < 3 \) tells that the weave pattern repeats after 6 positions. The essence is in a one-line Processing
(=Java) formula, embedded in the program as the 8th line.

```java
int i = 1000;
stroke(128);
boolean aij;
size(250,250);
for (int x=0 ;x<100; x++){
    int j = 0;
    for (int y = 0; y < 100; y++){
        aij = (i-j)%6<3 ? i%12<6 : j%12<6;
        fill(255 * int(aij));
        rect(10 * x, 10 * y, 10, 10);
        j++;
    }
    i++;
}
```

The program works with \( x \) and \( y \) coordinates which are multiplied by 10 to get black and white cells of
\( 10 \times 10 \) pixels. The variables \( i \) and \( j \) count warp threads and weft threads, respectively. We use a trick by
initializing \( i \) at 1000. This guarantees \( i-j \geq 0 \). Therefore the \% operator implements the mathematical \( \text{mod} \)
correctly. The formula belongs to a family of formulas, producing successive Pied de poule. In Figure 3 we
show four Pied de poule patterns, where the above formula generates the third pattern.

![Figure 3: Successive Pied de poule patterns.](image)

The leftmost pattern of Figure 3 is sometimes called Pied de poule, but in Dutch it is often called
“molenwiek” (the wings of a windmill). In English it is also called puppytooth. Later we shall come
back to the status of the “molenwiek”, whether it is a true Pied de poule or not. The general formula is
\( (i-j)\%2N < N? i\%4N < 2N : j\%4N < 2N \).

An alternative representation would be to use regular expressions e.g. \( \{0^61^6\}^* \) for a weft pattern. It is
at least as elegant and compact as \( i\%12 < 6 \), but the latter formula can go straight into the program (without
requiring an interpreter). The weaving part is not easily done by a regular expression, it is two-dimensional.
What happens when \( N \) becomes larger and larger? The basic figures\(^1\) in the Pied de poule become larger and larger (as shown in Figure 3), so eventually it becomes impractical to show them while keeping the cell size (weft thread width \( \times \) warp thread width) fixed. But we can generate a Pied de poule for large \( N \) and then scale back by a scaling factor, \( S \) say, inversely proportional to \( N \). This is how we made Figure 4 (left). Now we see that there is a limit Pied de poule. The size can be chosen arbitrarily, but the basic figure is fixed\(^2\). The largest Pied de poule pattern formally does not exist, but there is a limit case obtained as \( N \to \infty \).

![Figure 4](image)

**Figure 4**: Limit case Pied de poule pattern and M.C. Escher, *Birds E128*.

It is not practical to manufacture \( \frac{N}{N} \) woven Pied de poule for large \( N \) because the long thread loops make the fabric fragile. The patterns for large \( N \), and the limit Pied de poule pattern do find application in fashion nevertheless, but more sophisticated weaving techniques must be used; often they are printed.

### 3 tessellations

The limit pattern falls apart in pieces: the black pieces touch each other pairwise in one point. If we cut them at this point, we can identify a single black piece, which we simply call the “pied”. Inspired by this we can identify a pied for the patterns obtained for finite \( N \), for example \( N = 2, 3, 4 \). These are shown in Figure 5 (after scaling so they are of comparable size). Actually, the space left between the black areas has the same shape; these are the white pieds. In this section we shall study in more detail how the individual pieds can be fitted together again to fill the entire two-dimensional plane. First we notice that the pied for finite \( N \) is not entirely symmetric. The limit case pied is symmetric, however. Reflecting the limit case pied along the indicated diagonal line yields the same figure again (Figure 5). The pieds become ‘more symmetric’ when \( N \) becomes larger.

![Figure 5](image)

**Figure 5**: Successive pieds and limit pied.

A tessellation (or tiling) is a collection of figures that fill the 2D plane with no overlaps. They are used in textile design, interior design and industrial design [6]. M.C. Escher (1898-1972) added meaning to the tiles (e.g. animals), see Figure 4 (right) or [4, 5]. The mathematical theory of tessellation rests on two cornerstones: group theory, and topology. A geometric transformation which maps a figure to itself, is a symmetry of that figure. Certain transformations such as certain rotations, translations, reflections and

---

\(^1\)Note that we have no formal definition of “basic figure” yet; actually the concept is still ambiguous as it depends on how we choose to cut, but after further formalization, we shall see it is the interior of a certain closed line, as for example in Figure 9.

\(^2\)The limit also appears replacing integers \( i, j \) with reals \( x, y \). As one reviewer noted, this Mathematica code makes the limit pattern: \( \text{RegionPlot}\{\text{If}[\text{Mod}[(x - y), 2] < 1, \text{Mod}[x, 4] < 2, \text{Mod}[y, 4] < 2], \{x, 0, 8\}, \{y, 0, 8\}\} \).
combinations thereof, have been proven to be symmetries. A collection of operations including inverses (undoing the transformation) and compositions, is a group. In two dimensions one can create patterns filling the entire plane such that the pattern remains the same under certain transformations. For a specific pattern, this set of transformations forms a group, called a wallpaper group. There exist 17 wallpaper groups [7]. They appear in wallpaper, brickwork, floor tessellations, Eschers work, and indeed Pied de poule.

It is tempting to assume that each of the 17 regular wallpaper patterns gives rise to precisely one schema of regular tessellation. However, the relation is not one-to-one. What is still missing from the wallpaper patterns is: where are the cutting lines between the tiles? This question has been explored by Heesch and Kienzle and published in 1963 [8]. They added a topological analysis of the networks formed by the cutting lines between the tiles and gave 28 tessellation types, in a prescriptive style (in German). In Fig. 6 (left) and 7 (left) we show original descriptions of their first two tessellation types (the book is not widely available). The birds of Fig. 4 (right) follow the model Nr. 1 of Fig. 6 (left)3.

![Figure 6: Prescriptive model and Limit case Pied de poules tessellation](image)

Which Heesch-Kienzle type describes the Pied de poule patterns? Compare the regular patterns of Figures 3 and 4 to the models of Figure 6 (left) and 7 (left). Type Nr. 1 is in concordance with the tessellation of the limit case Pied de poule of Figure 4 (left). The tessellation is shown explicitly by omitting the color and showing the cutting lines, in Figure 6 (right) – we added the network points manually.

We expected that the patterns for finite $N$ form the same type of tessellation, but there is a surprise. The successive patterns for $N = 2, 3, 4$ of Figure 3 can not be described by Nr. 1. The point where four black and four white pieds come together just is not there. Actually we need Heesch-Kienzle type Nr. 2. This is applicable for the first three pieds of Figure 5 and in fact for any $N > 1$. Also see Figure 7.

4 Drawing the contour

We noticed that the cutting lines form an alternative characterization of Pied de poule. So what are these contour lines? Can we program them? For drawing lines we felt Turtle Graphics commands would be a good candidate. FD abbreviates FORWARD, RT is RIGHT and LT is LEFT. So RT 90 indicates turning clockwise over 90° and 1 is a distance. Asterisks are comments (interesting points).

```
LT 90 FD 1 * RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1
LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1
* FD 2 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1
RT 90 FD 3 LT 90 FD 2 RT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1
```
Turtle Graphics have been used in children’s mathematics education [10] and in 3D art [9]. In this way we get the line of Figure 8 (left), half the basic tile’s contour. We added the ‘interesting’ points manually. The small triangle gives the starting position and orientation of the turtle. Using a Speedy300 we laser cut the contours generated by a Turtle Command interpreter for $N = 3$. The pied has become a puzzle piece. A few of the 50 pieces we cut are shown in Figure 8 (center).

One can split the program above into three parts (the ‘*’ mark the splits). Then we can combine the command sequences for these parts with backward versions of the parts and thus draw the whole contour without lifting the pen. Yet the sequence of length 52 is unsatisfactory. The full contour is 104 (but from an information-theoretic viewpoint it is at most 52 plus a coding of the reversal and glue actions needed). But the list of 52 commands is neither compact, nor elegant, nor does it provide much insight. Re-arranging and subroutining the sequence is messy: Does a rotation belong to the next segment, or to the previous segment? Perhaps each rotation and one move could be packed together in a single command. We need a guiding principle. We found it by noting that the limit pied is bounded by straight lines belonging to three sets: horizontal, vertical and one direction of diagonal$^4$. See Figure 8 (right). So when a staircase has to be drawn, its procedure should be based on drawing one of these line types (adapted to the staircase situation).

We propose a variation of turtle graphics: we keep the idea of letting the turtle work relative to its own position, but the direction becomes absolute, rather than relative. It is a kind of sailor’s language: think of a sailor at sea with a chip log (also called common log or ship log) and a compass, but no clock or sextant. Such a sailor would know relative positions and absolute directions. Obvious commands are NORTH, SOUTH, EAST and WEST (and their abbreviated version $N$, $E$, $S$ and $W$). We call the command language Compass Logo.

Now we program functions for going South-East or North-West in a staircase-wise manner. For each we have two versions: ES means going east first and then south whereas SE means going south first and then

---

$^4$By algebra we find the lines replacing ‘<’ by ‘=’ in each component of $(i/S−j/S)%2N < N/4$, $i/S%4N < 2N$ scaling by $S=c/N$. The first is $(ciN−cjN)%2N = N$, solutions $i = j+1/c, i = j+2/c, …$ (slope $−45^\circ$, in Processing $y$ goes down).
east. We show them in Figure 9. Function definitions and conditionals are as in Berkeley Logo (UCBL), www.cs.berkeley.edu/~bh/logo.html.

Figure 9: Compact Compass Logo program for contour.

The same can be done for NW (North West) and WN (West North). The contour of the pied becomes straightforward, see the third and fourth box of Figure 9, whose output is in Figure 9 (right). We wrote an interpreter in Mathematica 8.0 as a functional program (in Scott-Strachey denotational semantics style). The interpreter is not efficient but usable. There is a general pattern NORTH $N-1$ ES $N-1$ E $N$ ES $2N$ W $N$ NW $N-1$ S $N-1$ W $N$ SE $N$ S $N$ WN $2N$ NORTH $N$ WN $N$ E $N$ where NORTH is the command, $N$ the parameter.

Recall the windmill or puppytooth given by Figure 3 (left). The turtle graphics approach of Section 4 provides us with a tool to explore this case. Fill in $N = 1$ (and remove useless commands such as NORTH 0). Then we find automatically the Compass Logo command sequence for the contour of the minimal pied, shown in Figure 10 (left). When we ran the program for the first time we expected to see the windmill pattern (Figure 10 center), but to our great surprise we got the rightmost drawing of Figure 10 instead. This was not what we expected. Yet we found no error and we came to realize that this is the minimal pied

Figure 10: Program for contour, naive expectation (left) and true minimal Pied (right)

Is the true minimal pied evidence that the term puppytooth is correct and is “windmill” wrong? First we must check for a third possibility: perhaps the windmills belong to another family of patterns as well? This is the case indeed (yet not explaining “windmill”). We found another family, viz. $a_{ij} = (i - j) \% 2 < 1 ? i \% 4N < 2N : j \% 4N < 2N$. For $N = 1$ it produces the same as Figure 3 (left) and for $N \geq 2$ the checkers shown in Figure 11 (called Pepita; used in chefs trousers and Porsche car interiors, amongst others). Again there is a limit pattern. In the limit we get squares areas with luminosity being the average of the areas with 50% black and 50% white squares: gray (it is unclear what the basic figure is). Figure 11 gives the family structure Venn-diagram. To our surprise, the “windmill” does not stand out as something common to the checkerboard family. The term “windmill” refers to a Gestalt which exists for $N = 1$ only.

Figure 11: Venn diagram of Checkerboard and Pied poule patterns.
5 New tessellations

Playing with the laser cut tiles we noticed that they interlock for $1 < N < \infty$, see Figure 8 (center). Once head-tails interlocked, they cannot be pulled apart without lifting. The tessellation has infinitely long staircase cutting lines; it falls apart in strips of interlocked pieds, which can be shifted with respect to each other.

![Figure 12: Model Nr.7 with translations (T) and 180° rotations centers (C) and novel "pied de poule".](image)

First we guessed that there exists only one tessellation type (since the pied can be obtained by following the prescription Nr. 2). But playing with the strips more tessellations appear, based on translations along the infinite staircase cutting lines. In this way we find $2N - 1$ regular tessellations. We can also produce irregular tessellations (since the translations can be different at each line). We included rotations: rotate strips by $180^\circ$ alternatingly. This fits because there are additional symmetries hidden in the sides of the basic pied which distinguish it from a Heesch-Kienzle tile which is really arbitrary (“willkürlich”). An additional tessellation is given in Figure 12 (right). The contour has line segments which are invariant with respect to $180^\circ$ rotation around certain centers (the long staircase segments and the tail ends). This is a Nr. 7. The tessellation of Figure 12 (right) is made of pieds for $N = 3$. The connection points and the rotation centers are indicated. This pattern is new to the best of our knowledge. The small rightmost figure shows a black and white pattern of this type for $N=2$. This does not exhaust the possibilities for creating novel patterns, such as adding reflection of entire strips. We leave this as an option for future research.

Acknowledgements: I like to thank Christoph Bartneck, Jun Hu, and Tom Verhoeff for the cooperation in creative programming, Katinka Feijs for the fashion inspiration, and the reviewers for valuable feedback.

References


---

Fig. 12 (left) With kind permission from Springer Science+Business Media: Flächenschluss; System der Formen lückenlos aneinanderschliessender Flachteile. Heesch, H., & Kienzle, O., 1963, Fig. TCCTCC Nr 7, p.67.