The Cartographic Journal *Vol. 47 No. 3 pp. 257–261 August 2010*

REFEREED PAPER

Point Pattern Synthesis

Bernhard Jenny, Ernst Hutzler and Lorenz Hurni

Institute of Cartography, ETH Zurich, Zurich, Switzerland
Email: jenny@karto.baug.ethz.ch

Patterns consisting of point symbols are commonly applied to symbolize land cover. Tree symbols, for example, indicate forested areas; small irregular dots show quarries and pits; or regular line symbols represent a vineyard. This paper presents an automatic method for synthesizing patterns consisting of one or more point symbols. Symbols of varying sizes or graphical appearance can be combined according to user-defined ratios. The automatic method generates patterns of regularly or irregularly distributed symbols. It ensures that symbols do not overlap and do not graphically conflict with other map features. The method has been implemented in a free plug-in for the Adobe Illustrator vector graphics editor.

Keywords: proportional point symbol pattern, vector texture synthesis

INTRODUCTION

Filling polygons with small symbols is an effective cartographic technique for symbolizing land cover or other area-related nominal variables. Patterns are built up from a single type of point symbol or a mixture of multiple point symbol classes. For example, uniform dot symbols can represent an orchard, or large and small circles can be combined to represent scrubs and scattered trees (Figure 1).

The synthesis of point patterns differs from dot mapping, a technique commonly used to communicate variation in spatial density (Dent, 1999; Slocum *et al.*, 2009). In a dot map, one symbol represents one or multiple items to illustrate a spatially varying variable. Hence, the density of dots varies throughout the map, and map readers can derive quantitative information by counting the dot symbols.

In contrast to dot maps, point patterns show an approximately uniform density of point symbols in space, and a single symbol does not represent an absolute quantity. Instead, point patterns represent either nominal variables, or regionally varying ratios. Nominal variables are far more common, both in topographic maps (e.g. land cover patterns as in Figure 1) and thematic maps (e.g. the distribution area of an animal species). Regionally varying ratios are the less frequent variant. Ratios can be communicated by patterns consisting of two or more different symbols by varying their relative number among area units. For example, the ratio of stylized fir symbols versus oak tree symbols can indicate the variation in deciduous and coniferous trees among forest patches. This technique for mapping ratios can be considered a special case of multivariate dot maps that use a distinct shape or colour for each attribute (Slocum *et al.*, 2009, p. 337; Rogers and Groop, 1981).

The spatial distribution of symbols in a pattern can vary between a regular grid-like alignment and a completely random distribution. The kind of distribution usually indicates characteristics of the symbolized feature. For example, symbols for vineyards and orchards are often arranged in checkerboard patterns to mimic the regularity of the land cover represented, while scattered circles indicate dispersed bushes on scrubland (Figure 1). Spiess (1988) provides additional principles for the design of area patterns.

POINT PATTERN SYNTHESIS METHODS

There are two different groups of algorithmic approaches for the synthesis of point patterns. The first group of methods start from a user-defined reference pattern that is analysed and repeated in a pseudo-random manner to fill an area (Figure 2). The basic concept behind these synthesizing methods is very simple; however, algorithms can become highly complex as the symbols forming the pattern must be placed and rotated individually to replicate the artist's style in the reference pattern. Indeed, pattern synthesis based on sample patterns is a field of active research in computer graphics (Barla *et al.*, 2006; Hurtut *et al.*, 2009; Ijiri *et al.*, 2008).

An alternative group of methods places predefined point symbols pseudo-randomly inside an area to be filled. A commonly used method is based on jittered grids, which is computationally efficient and simple to implement. It starts with a regular grid, places a symbol on each grid node and then adds a random displacement to the position of each symbol, whereby the maximum displacement is limited, for instance, to half a grid cell. This method has also been used for automated dot mapping, for example, by Lavin (1986)

DOI: 10.1179/000870410X12786821061378

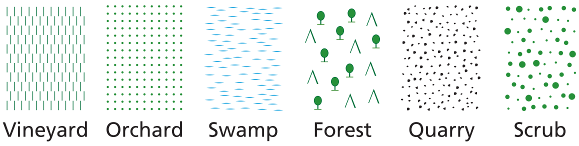

Figure 1. Regular and irregular patterns of one or multiple point symbols

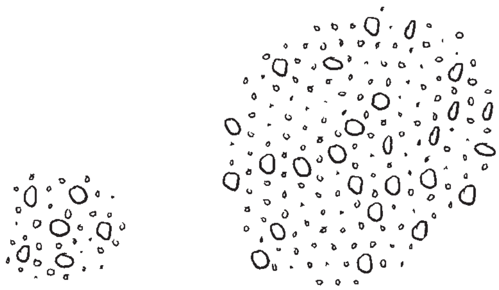

Figure 2. A small sample (left) for synthesizing a large pattern (right) (Barla *et al.*, 2006)

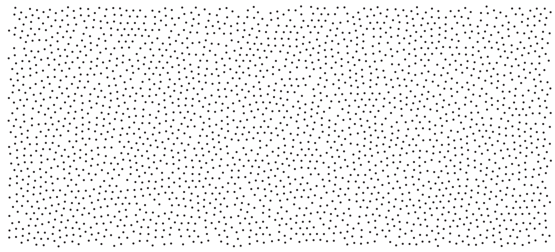

Figure 3. Poisson-disk distribution (Dunbar and Humphreys, 2006)

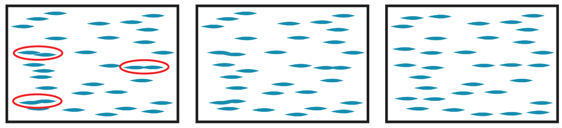

Figure 4. Overlapping symbols create graphically salient conflicts (left and middle); a more even pattern results when overlaps are corrected (right)

and Ditz (1999). The jittered grid method is commonly used in ray tracing for stochastic sampling to generate anti-aliased images and special effects (Cook, 1986). More sophisticated methods for synthesizing stochastic sampling patterns have been developed in computer graphics, aimed at approximating the Poisson-disk distribution. The Poisson-disk distribution consists of points that keep a minimum distance between them, and mimics the distribution of the photoreceptors in a human eye (Yellot, 1983). Poisson-disk distributions, therefore, have a natural look and are pleasant to look at (Figure 3). However, their disadvantage compared to jittered grids is the considerable algorithmic complexity and the slower performance of the algorithms. As a matter of fact, Poisson-disk distributions have proven difficult to generate directly, so many alternative approaches have been developed (for a literature overview, see Dunbar and Humphreys, 2006).

Owing to their pleasant graphical qualities, one could argument that the Poisson-disk distribution should be used for irregular point patterns. However, most point patterns on maps do not cover large areas and are intermingled with other map features, such that the pleasant graphical qualities of the Poisson-disk distribution often do not become apparent to the map user. Furthermore, if a point pattern consists of multiple symbols that vary considerably in size, the synthesis methods based on sample patterns might result in more homogenous symbol distributions. However, as such methods are complex and research is ongoing, a variant of the simpler grid jittering is used for the study presented in this paper.

POINT PATTERNS WITHOUT GRAPHICAL CONFLICTS

The method proposed here for cartographic pattern synthesis starts by placing point symbols at nodes of a regular grid. The spacing of the grid, as well as the maximum random displacement of each symbol, can differ in both horizontal and vertical directions. In Figure 4, for example, vertical spacing between symbols is smaller than horizontal spacing to account for the elongate symbols, and the horizontal jittering amount is larger.

Point symbols in area patterns usually do not overlap other map features, but are placed at a certain minimum distance. Symbols lying too close to other features would result in unaesthetic patterns that are ambiguous and difficult to read. For example, point symbols should not be placed on top of or below a crossing footpath. In addition, self-overlapping among symbols of the same type is generally avoided, also for reasons of aesthetics and readability (Figure 4). Hence, in order to make rendering labour-intensive interactive corrections unnecessary, an automatic method is required to solve such graphical conflicts.

The following paragraphs extend pattern synthesis based on jittered grids with techniques that automatically solve these types of conflicts. The user can control the final pattern with a few numerical parameters. The proposed method starts with a regular grid distribution of symbols (Figure 5a), and applies a random displacement to each symbol (Figure 5b). The maximum amount of the displacement is half the cell size of the grid, or less, as specified by the user. The algorithm then identifies graphical conflicts among symbols and other map features. If a conflict exists, it is solved by displacing (Figure 5c) or removing a symbol (Figure 5d). This procedure of conflict detection and solution is executed in an iterative sequence. The iterations stop when no conflicts remain or when no further improvements are possible. Conflict solving by removing point symbols may generate visually salient gaps in the pattern. These gaps are filled in a final processing step by inserting additional symbols (Figure 5e).

Hence, three types of graphical conflicts exist: (1) symbols can overlap one another, or symbols can be placed too close to one another; (2) symbols can overlap other map features, or undershoot the minimum distance to other features; and (3) symbols can be too close to the edges of the area that is filled with the pattern.

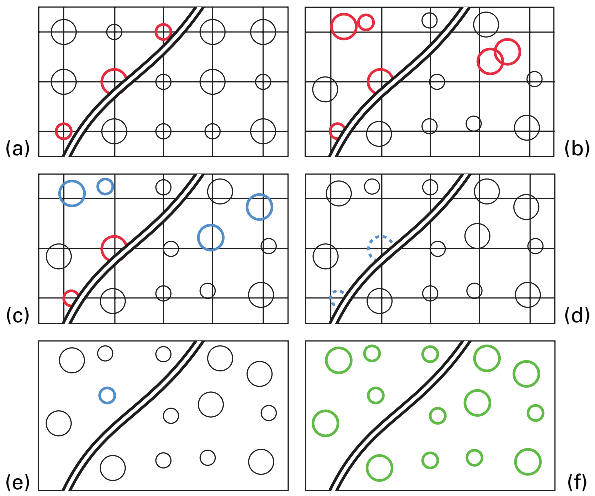

Figure 5. Random distribution of point symbols (conflicting symbols in red and changes in blue): (a) initial regular pattern; (b) random displacement; (c, d) displacement and removal for conflict solution; (e) filling the gaps by inserting symbols; (f) final pattern

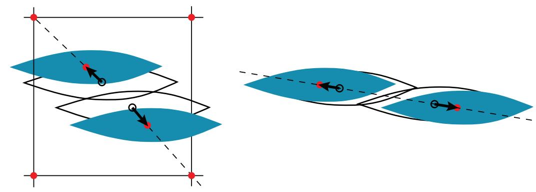

Figure 6. Solving conflicts between overlapping symbols: moving towards the nodes of the generating grid (left) or along the direction defined by the current centres

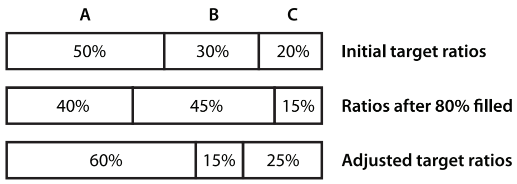

Figure 7. Dynamic adjustment of class proportions while placing symbols

(1) Overlapping symbols or symbols too close to one another

Jittering can result in symbols overlapping or lying too close to one another, especially when the maximum random displacement is large compared to the symbol size (Figure 4). We have explored two methods to solve this type of conflict. The first method moves the symbols towards the nodes of the generating grid (Figure 6, left). This is the preferred solution if space is limited (i.e. symbols are dense) because it does not tend to generate further conflicts with other symbols. The second method moves the symbols away from one another, in the direction defined by the centres of their current positions (Figure 6, right). This solution is preferred if there is more space available as it results in a more irregular distribution.

(2) Symbols overlapping other map features or undershooting the minimum distance to other features

For most patterns, symbols should keep a minimum distance to other map features, such as a road that crosses the pattern area. The border of the conflicting map feature could be used to mask out the point symbols, but this would result in partially cropped symbols. Point symbols would be degraded to unrecognisable and graphically unpleasing particles (Spiess, 1988). A preferable approach consists in first detecting the conflicts, then either removing or displacing the conflicting symbols.

(3) Symbols located too close to the edges of the polygon containing the pattern

This conflict is graphically and logically different from the first two conflicts described above. However, the solution is identical to the case where symbols overlap other map features. If overlaps exist, a symbol is removed or displaced. However, the algorithm is simpler as only conflicts with the bounding polygon have to be detected.

The three types of conflicts described require frequent distance computations among symbols, or between symbols and other map features. We found that distance computations based on the symbols' rectangular bounding boxes provide sufficiently accurate results in most cases. However, it is essential to take the stroke widths of vector art into account. Otherwise, distances are consistently overestimated and some conflicts remain undetected.

In order to accelerate the distance computations, invisible auxiliary grid structures are used in our implementation. These auxiliary grid structures are a simple, yet efficient means for quickly finding map features that are close to each another. Each cell of the auxiliary grid contains references to overlapping map features. Once initialized with existing map features, it is quick to find all features positioned within a certain distance to a symbol. The spatial search algorithm merely iterates over the grid cells around a symbol for detecting neighbouring features within a certain distance, or when scanning for gaps in the pattern that need to be filled with additional point symbols.

MAPPING RATIOS WITH POINT PATTERNS

To generate patterns consisting of multiple point symbols, the user specifies the relative number for each symbol class by fractions (e.g. 50% of all symbols of class A, 30% of class B and 20% of class C). The algorithm then randomly selects the symbols from the classes. This selection is constrained, such that the symbols approach the correct fractions for a large number of samples. However, the resulting pattern is unlikely to contain the exact fractions of each symbol class, due to the relatively small number of symbols placed and the random selection of the symbol classes. An additional reason is that some symbols are removed from the pattern during the iterative process described above, if they conflict with other map features. Hence, a priori it is not possible to compute the number of symbols to be placed inside an area.

To achieve a distribution that approximates the specified fractions, our pattern synthesis method automatically adjusts the probability for selecting a certain class after a determined number of symbols have been placed. The first few symbols are sampled from the classes, and the number of symbols actually placed in the pattern is recorded for each class. After a certain number of symbols are placed, the probabilities of each class are adjusted. If fewer symbols of a class were placed than required by the specified fractions, the probability for this class is increased, and *vice versa*.

The method is best explained by means of a numerical example: For a pattern consisting of three types of symbols, the user wants 50% of all points of symbol class A, 30% of symbol class B and 20% of symbol class C (Figure 7, top row). When adding the symbols to the pattern, a pseudo-random procedure is used. For each symbol, a random number is generated between 1 and 100. If this random number is below 50, a symbol of class A is added to the pattern; if the number is between 50 and 80, the symbol added is of class B; and if the random number is above 80, the symbol is of class C.

After 80% of the pattern area is filled, the algorithm computes the relative numbers of symbols added to the pattern. In the example, 40% of all symbols placed are of class A, 45% of class B and 15% of class C (Figure 7, middle row). The ratios are then adjusted for the forthcoming symbols for the remaining area by adding the difference between the targeted and the real ratio. In the example, the adjusted probability of class A is computed with: 50%+(50%–40%)=60%. The other probabilities are equally adjusted, resulting in 15% for class B and 25% for class C (Figure 7, bottom row). In our implementation, the fractions are not only adjusted after 80% of the pattern area is filled, but also after 90 and 95%, which augments the probability for the final distribution to approach the targeted ratios. However, it is important to note that for small patterns consisting of only a few symbols, the final ratios almost never match the target ratios exactly. In addition, the described procedure for adjusting the ratios is unnecessary for patterns consisting of a few hundred symbols as the adjustment is unnoticeable to the human eye.

IRREGULAR PATTERN PLUG-IN

The described techniques for generating point patterns were implemented in a plug-in for Adobe Illustrator CS4, a vector graphics editor used by many cartographers. It is part of a set of specialized plug-ins for cartographers developed at the Institute of Cartography of ETH Zurich (Hurni and Hutzler, 2008; Werner and Hutzler, 2006). The Irregular Pattern plug-in offers a dialogue for selecting the symbols to be placed, their relative share and their spatial distribution, as well as the minimum distances to other map features, the enclosing polygon edges and other symbols (Figure 8).

DISCUSSION

The described method for the automatic generation of point patterns offers the following advantages from a user's point of view: ratios for the combination of symbols of varying sizes or graphical appearance can be specified; symbols are optionally displaced forming an irregular pattern without overlapping; and symbols do not coalesce with other map features.

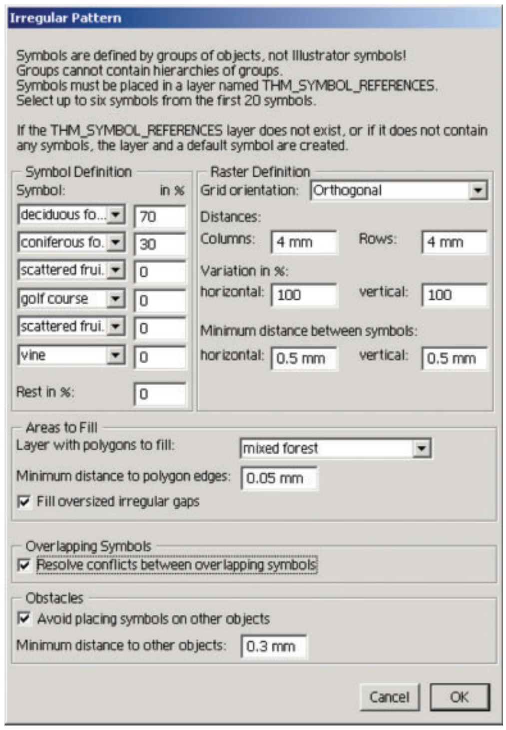

Figure 8. Screenshot of the Irregular Pattern plug-in for Adobe Illustrator

However, improvements to the method are conceivable. For more accurate distance computations, rather than the rectangular bounding box of symbols, a convex hull or exact geometry could be used. Moreover, the minimum distances to other map features and to point symbols are currently identical for all point symbol classes. A more flexible approach would allow the user to define individual distances for each symbol class. Additionally, the size and density of symbols could be modulated with other variables, an approach adapted for digital scree mapping for topographic maps of high-mountain areas, where the density of scree dots varies with the grey values of a shaded relief (Jenny *et al.*, 2010). Finally, the absolute number of symbols placed could be defined per polygon to produce quantitative dot maps.

The jittered grid method is not suitable if very large symbols and very small symbols are combined, since all symbols are initially distributed along a regular grid. This results in relatively large distances between small symbols. More advanced methods based on reference patterns, as outlined above, might yield more appropriate results for these cases. However, patterns consisting of symbols with extreme differences in size are relatively rare in cartography.

Methods for synthesizing Poisson-disk distributions could possibly result in graphically more pleasing distributions when large areas are filled. However, most pattern areas are intermingled with other map features, often hiding the pleasant look of Poisson-disk distributions.

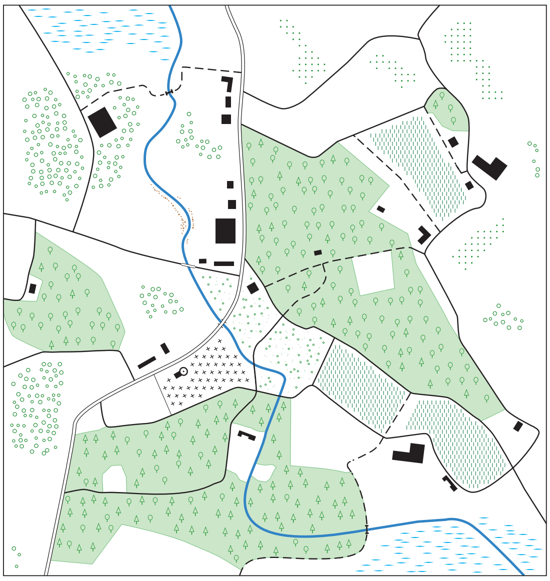

Figure 9. Various types of patterns generated with the Irregular Pattern plug-in

Figure 9 shows a topographic map containing various types of point patterns generated with the method presented in this paper. The patterns in the map were generated using the Irregular Pattern plug-in for Adobe Illustrator. Pattern generation is quick, and vector symbols of arbitrary geometry can be placed. The Irregular Pattern plug-in provides sufficient control options for most patterns and is available free for the Windows operating system at http://www.ika.ethz.ch/plugins/.

BIOGRAPHICAL NOTES

Bernhard Jenny is a post-doctoral researcher at the Institute of Cartography, ETH Zurich, where he also did his PhD studies. He studied geomatics, surveying and environmental science at EPFL Lausanne, Switzerland, and also holds a post-graduate certificate in computer graphics from ETH Zurich. His research interests include, among others, cartographic terrain representation, web mapping, the visualisation and analysis of map distortion, and the design of world map projections.

Ernst Hutzler studied communications engineering at the University of Applied Sciences in Konstanz, Germany, and is now a software developer at the Institute of Cartography, ETH Zurich. He specialized in the development of cartographic extensions and plug-ins for vector graphic software, such as Adobe Illustrator. His plug-ins for Adobe Illustrator (including the Irregular Pattern plug-in) are available free on the homepage of the Institute of Cartography, ETH Zurich.

Lorenz Hurni is Professor at the Institute of Cartography, ETH Zurich. He is the editor-in-chief of the *Atlas of Switzerland* (the Swiss national atlas), and the *Swiss World Atlas* (the official school atlas). His current research focus is on cartographic data models, tools for the production of printed and multimedia maps, as well as interactive, multi-dimensional multimedia map representations.

ACKNOWLEDGEMENTS

The authors would like to thank Michael Borop of *World Sites Atlas* for writing a description of our plug-ins in English. We would also like to thank Annemarie Jordan, TU Dresden, for her help with the figures in this publication and in the plug-in manual.

REFERENCES

Barla, P., Breslav, S., Thollot, J., Sillion, F. and Markosian, L. (2006). 'Stroke pattern analysis and synthesis', **Computer Graphics Forum**, 25, pp. 663–671.

Cook, R. L. (1986). 'Stochastic sampling in computer graphics', **ACM Transactions on Graphics**, 5, pp. 51–72.

Dent, B. D. (1999). **Cartography – Thematic Map Design**, 5th ed., WCB McGraw-Hill, Boston, MA.

Ditz, R. (1999). 'The visualization of population distribution in a cartographic information system – aspects of technical realization of dot maps on screen', in **19th International Cartographic Conference**, pp. 965–972, Ottawa, Ont., Aug 14–21.

Dunbar, D. and Humphreys, G. (2006). 'A spatial data structure for fast Poisson-disk sample generation', **ACM Transactions on Graphics**, 25, pp. 503–508.

Hurni, L. and Hutzler, E. (2008). 'Implementation kartographischer Funktionen in Adobe Illustrator-Plugins', **Kartographische Nachrichten**, 58, pp. 248–255.

Hurtut, T., Landes, P., Thollot, J., Gousseau, Y., Drouillhet, R. and Coeurjolly, J. (2009). 'Appearance-guided synthesis of element arrangements by example', in **Proceedings of the 7th International Symposium on Non-Photorealistic Animation and Rendering**, ed. by Spencer, S. N., pp. 51–60, ACM, New York.

Ijiri, T., Mech, R., Igarashi, T. and Miller, G. (2008). 'An example-based procedural system for element arrangement', **Computer Graphics Forum**, 27, pp. 429–436.

Jenny, B., Hutzler, E. and Hurni, L. (2010). 'Scree representation on topographic maps', **The Cartographic Journal**, 47, pp. 141–149.

Lavin, S. (1986). 'Mapping continuous geographical distributions using dot density shading', **The American Cartographer**, 13, pp. 140–150.

Rogers, J. E. and Groop, R. E. (1981). 'Regional portrayal with multi-pattern color dot maps', **Cartographica**, 18, pp. 51–64.

Slocum, T. A., McMaster, R. B., Kessler, F. C. and Howard, H. H. (2009). **Thematic Cartography and Geovisualization**, 3rd ed., Pearson Prentice Hall, Upper Saddle River, NJ.

Spiess, E. (1988). 'Map compilation', in **Basic Cartography for Students and Technicians**, ed. by Anson, R. W., Vol. 2, pp. 23–69, ICA, London.

Werner, M. and Hutzler, E. (2006). 'Contemporary map design with desktop-publishing system extensions', **Wiener Schriften zur Geographie und Kartographie**, 17, pp. 224–232.

Yellot, J. I. (1983). 'Spectral consequences of photoreceptor sampling in the rhesus retina', **Science**, 221, pp. 382–385.