Investigations of Game of Life cellular automata rules on Penrose Tilings: lifetime and ash statistics

Nick Owens[1] and Susan Stepney[2]

[1] Department of Electronics, University of York, UK
[2] Department of Computer Science, University of York, UK

Abstract. Conway's Game of Life rules can be applied to Cellular Automata (CAs) running on aperiodic grids, namely Penrose tilings. Here we investigate the result of running such CAs from random initial conditions. This requires development of a Penrose tiling algorithm suitable for CA experiments, in particular, a tiling that can be lazily expanded as CA activity reaches an edge. We describe such an algorithm, our experimental setup, and demonstrate that the Penorse kite and dart tiling has significantly different statistical behaviour from the Penrose rhomb tiling.

1 Introduction

John Horton Conway's *Game of Life* [3][8] is a simple two-dimensional, two state cellular automaton (CA), remarkable for its complex behaviour [3][16]. That behaviour is known to be very sensitive to a change in the CA *rules*. Here we continue our investigations [11] into its sensitivity to changes in the *lattice*, by the use of an aperiodic Penrose tiling lattice [9][14].

Section 2 reviews Penrose tilings, and section 3 describes algorithms to generate them, including one that permits 'lazy extension' of the tiling. Section 4 generalises the concepts of neighbourhood and totalistic CA rules to aperiodic lattices. Section 5 describes the experimental setup for running the Game of Life rules on aperiodic lattices; section 6 reports the statistics of lifetimes, ash densities, and growth of the region of activity.

2 Penrose tilings

2.1 Kites and darts, and rhombs

Grünbaum & Shephard [10, chapter 10] provide a good introduction to aperiodic tilings, including Penrose tilings. The two variants of Penrose tiling we consider here are 'kites and darts', and 'rhombs'.

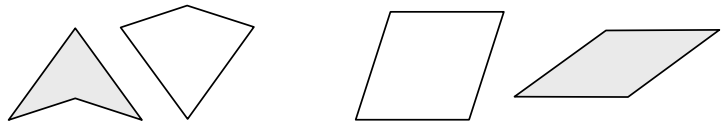

Fig. 1. Penrose tiles (a) the dart (grey) and kite (white) tiles: the long and short sides are in the ratio $\phi : 1$, where the golden ratio $\phi = (1 + \sqrt{5})/2 = 2\cos(\pi/5)$; (b) the thick (white) and thin (grey) rhomb tiles

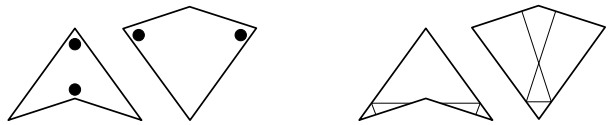

Fig. 2. Kite and dart matching rules (a) vertex markings (b) Ammann bars

The kite and dart tile pair are shown in Fig. 1a; a large patch of kite and dart tiling is shown in Fig. 19. The thick and thin rhomb tile pair are shown in Fig. 1b; a large patch of rhomb tiling is shown in Fig. 21. [21] shows that the number of thick to thin rhombs in a Penrose tiling is in the ratio $\phi : 1$.

2.2 Matching rules and Ammann bars

To avoid a kite and dart being joined to form a rhombus (Fig. 7), and hence a periodic tiling, there are additional 'matching rules': as well as edges of the same length being put together, certain vertices (given by the dots in Fig. 2a) must also be matched [9][10]. Another matching scheme uses Ammann bars (Fig. 2b), which must be joined in straight lines across tiles [10]. (Completed Ammann bars highlight the underlying structure, and can be used to construct tilings; see section 3.5.)

To avoid rhomb tiles being used to form a periodic tiling, there are additional 'matching rules': as well as edges of the same length being put together, the edge orientations (given by the arrows and dots in Fig. 3a) must also be matched [5]. Another matching scheme uses Ammann bars (Fig. 3b), which must be joined in straight lines across tiles.

2.3 Valid vertex configurations

There are many ways to put the tiles together, even with the restriction of the matching rules. However, in a true Penrose tiling (one that can be extended to infinity), not all of these configurations can exist.

There are only seven valid ways to surround any vertex in a kite and dart tiling [9] (Fig. 4).

There are only eight valid vertices in a rhomb tiling [5] (Fig. 5). The names of these vertices come from the names of the corresponding kite and dart vertices

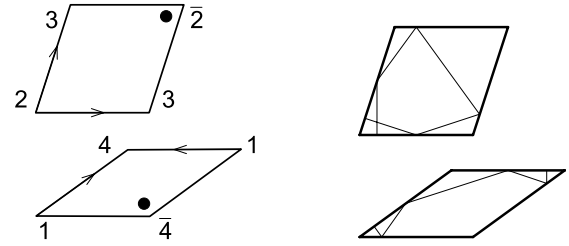

Fig. 3. Rhomb matching rules (a) vertex marking and edge orientations plus vertex angle numbering, where interior angles are $\pi/5$ times the vertex angle number (note that vertices labelled 2, and labelled 4, come in two kinds, due to the matching rules: these are distinguished by overbars) (b) Ammann bars

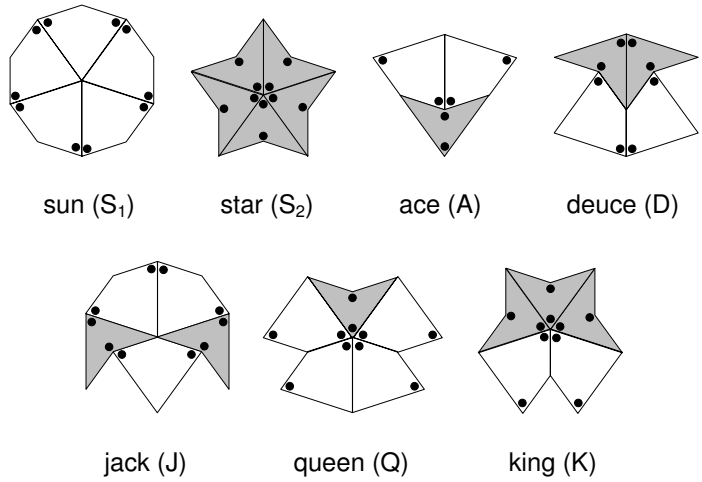

Fig. 4. The seven valid vertex configurations of a kite and dart tiling [9]

from which they can be derived [5]. Each vertex can be associated with a list of vertex angle numbers (after [17, fig.6.8], augmented here with overbars, Fig. 3a), corresponding to the vertex angles of the tiles forming the central vertex. These are useful for determining how to complete forced vertices (see section 3.3). Note that there are two distinct occurrences of the 3,3 vertex configurations (in the J and D); see Fig. 6.

If a patch of tiling exhibits any other vertex configuration, it is not a true Penrose tiling: it will not be possible to extend it to infinity. We use these valid vertex configurations to analyse valid neighbourhood configurations later.

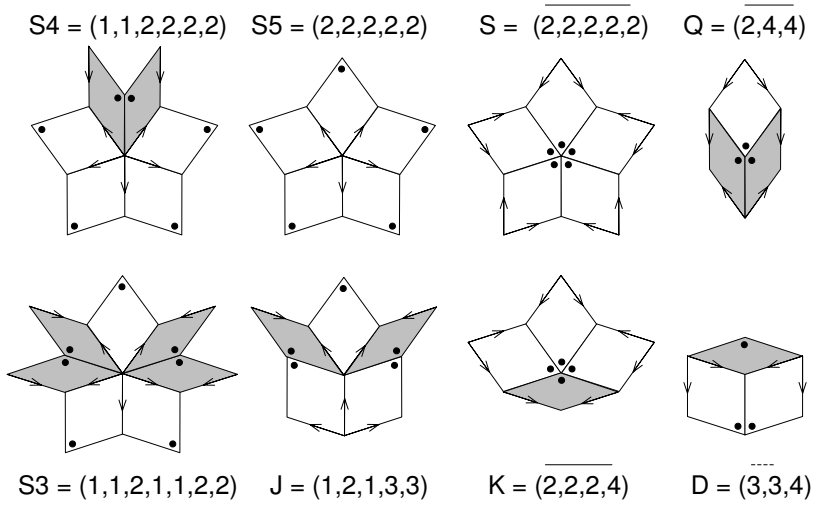

Fig. 5. The eight valid vertex configurations of a rhomb tiling [5]

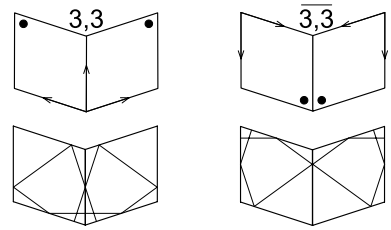

Fig. 6. Disambiguating the 3,3 vertices: the two distinct ways a 3,3 vertex can appear in a valid rhomb vertex configuration (in the J and D, see Fig. 5). This is a *context dependent* marking [21].

3 Penrose tiler construction

3.1 Requirements

There are several different algorithms for constructing valid Penrose tilings of arbitrary size. Here our aim is to experiment with CA rules on a Penrose tiling. We need to decide what to do as activity approaches the edge of the current grid. One common solution to this is to implement a 'lazy' grid, which expands as necessary. This provides requirements for a Penrose tiling algorithm suitable for investigating CAs:

1. The lazy tiler shall tile an arbitrarily sized rectangular region with a valid Penrose tiling (supporting both kites and darts, and rhombs)

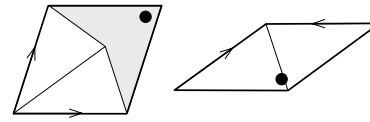

Fig. 7. Relationship between (marked) rhomb tiles and kites and darts: a thick rhomb comprises a dart and two half-kites; a thin rhomb comprises two half-kites

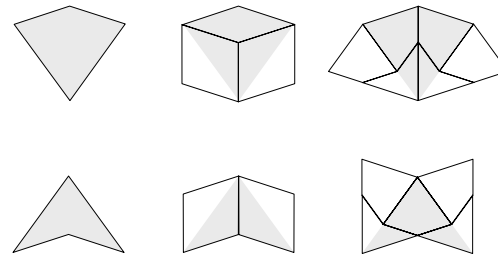

Fig. 8. One iteration of deflating a dart tile and a kite tile, via intermediate rhomb tiles (after [10, Figure 10.3.19]). The ratio of the original and final kite and dart tile sizes is the golden ratio ϕ.

2. Given any point on the plane outside the tiled region, the tiler shall appropriately expand the tiled region to a larger rectangle that includes this point
3. During such expansion, the tiler shall not extend tiling unnecessarily (neither in directions away from the expansion point, nor beyond it)
4. The tiler shall determine the neighbourhood of each tile, for use as a CA lattice
5. The tiler shall determine when CA activity reaches the edges of the currently defined lattice, and trigger suitable grid expansion

Our pentagrid lazy tiling algorithm meets these requirements. We describe it here in some detail, because previous descriptions of pentagrid tiling algorithms are somewhat obscure (and not tuned for CA experiments), and the lazy extension algorithm has not been described before.

3.2 Deflation

The relationship of the rhomb tiles to the kite and dart tiles is shown in Fig. 7. The deflation tiling method involves recursively breaking tiles into sub-tiles (Fig. 8). This is one of the best known method of creating a valid Penrose tiling, and derives from methods used in [14].

Ramachandrarao *et al* [15] describe a related decomposition process that produces a valid rhomb tiling starting from a single thick rhomb, where the decomposition processes are "akin to normal crystallographic operations".

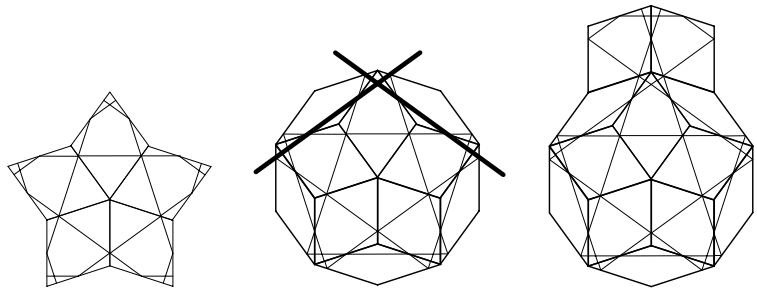

Fig. 9. Example of Onoda *et al* [13] rhomb tiling algorithm. (a) Start from an S vertex initial seed cluster; each concave vertex on the tiling edge is $(\bar{3}, \bar{3})$, so is a forced D vertex. Apply *R1* to completed each with a thin rhomb. (b) The result is a regular decagon, with no further forced vertices. The underlying Amman bars intersect at 108° (bold lines). *R2* requires a thick tile consistent with the vertex rules to be added "*to either side*" (that is, to one side or the other) of the corresponding tiling vertex, which here has (partial) vertex number list (1,2,1) (c) Both S3 and J have a partial vertex list of (1,2,1), but extension with a *thick* tile implies extension with a 2, 3 or $\bar{3}$ vertex. The only way this can be done consistently is to complete it as a J vertex, (1,2,1,3,3).

The deflation approach was taken in [11]; as noted there it is not a suitable process for a lazy tiler, since the generation n grid does not clearly appear as a subpart of the larger generation $n + 1$ grid. So although it is possible to create arbitrarily large tilings (requirement 1), it is not possible to do so 'lazily' (requirement 2); this size must be set at the start.

3.3 Onoda's rhomb tiling algorithm

Onoda *et al* [13] describe an algorithm for generating rhomb tilings. Start from any valid 'seed cluster' tiling (for example, a single tile, or one of the eight valid vertex tilings in Fig. 5). "*R1: If one or more vertices are forced, choose a forced vertex and add a forced tile to it.*" A forced vertex is one that can be completed in only one way to give a valid vertex. (Without loss of generality, we can add the entire set of tiles needed to complete the forced vertex.) "*R2: If there are no forced vertices, add a thick tile (consistent with the vertex rules) to either side of any* 108° *corner.*" The 108° corner referred to in this rule is not (necessarily) a 3-vertex of a thick rhomb: it is rather the underlying corner defined by the edges of the completed forced patch of tiling "*when viewed macroscopically*". These macroscopic edges are traced out by the underlying Amman bars in the matching rules (Fig. 9). Again, without loss of generality, we can add the entire set of tiles needed to complete the chosen vertex.

This process results in a steadily growing patch of tiling (requirements 1 and 2). However, it is not suitable for a lazy tiler, since the direction that the tiling

grows is not under programmatic control: it has to grow in the direction of forced vertices first, then 108° corners (so does not satisfy requirement 3).

3.4 Ammann bars

The underlying Ammann bars of a Penrose tiling form five grids of lines, each grid rotated by a multiple of $2\pi/5$ (Fig. 9). Given such a grid, the underlying tiling can be reconstructed.

A lazy tiler based on Ammann bars could satisfy requirement 3, as new grid lines could be laid down only in the required direction. However, these grid lines are not evenly spaced: they form a Fibonacci sequence [21]. The lazy tiling extension (R2), which involves laying down new grid lines, would be possible, but slightly tricky. Since there is a similar approach, but based on a regularly-spaced pentagrid (see next section), the Ammann bar approach is not considered further here. ([21] shows that if one starts with a rhomb tiling reconstructed from Ammann bars, and deflates it, then the resulting rhomb tiling is one that can be reconstructed from a regular pentagrid.)

3.5 Pentagrid

A multigrid comprises sets of evenly spaced parallel lines all at appropriate rotations of each other. de Bruijn [6] shows that a multigrid is the dual of a tiling. (The dual tiling of a grid associates a tile with every intersection of the grid, and a vertex of the tile with every open area of the grid.) A special case of the multigrid is the *pentagrid*, and its dual is a Penrose rhomb tiling [5].

The description of how to extract the dual rhomb tiling from a pentagrid given in this section is adapted and simplified from [5] [21]. de Bruijn [5] provides the original pentagrid definitions and theorems. His formulation is in terms of complex numbers (rather than vectors), and has no diagrams, which can make it hard to understand the approach in algorithmic terms. Socolar & Steinhardt [21] recast the approach in terms of vectors, but also generalise to multigrids and three dimensions, which again can obscure the components of a tiling algorithm. Austin [1] provides a good overview of the technique.

A *grid* (Fig. 10a) is a sequence of regularly spaced parallel lines, labelled with the integers. A unit vector $\mathbf{e}$ normal to the lines defined the direction of the grid. A number γ defines the distance of the line labelled zero from the origin. So the grid line labelled N is defined by

$$\mathbf{x}.\mathbf{e} = N + \gamma \tag{1}$$

where N is the line label.

A point $\mathbf{x}'$ not on a grid line is assigned an index value equal to the higher label of the adjacent grid lines:

$$N_{x'} = \lceil \mathbf{x}'.\mathbf{e} - \gamma \rceil \tag{2}$$

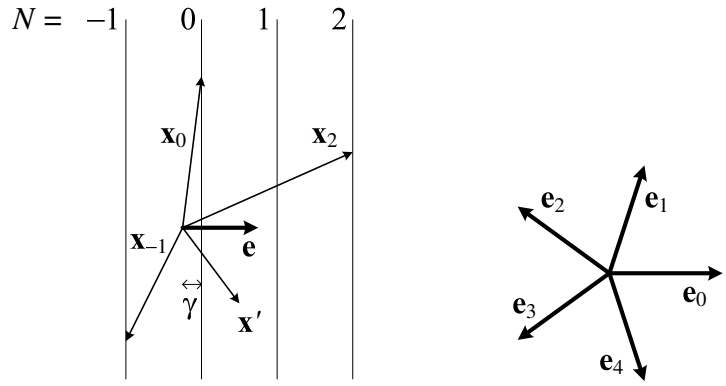

Fig. 10. (a) A grid defined by unit vector $\mathbf{e}$ and offset γ (b) Five unit vectors, with angles $2k\pi/5 = k \times 72°$, defining a pentagrid. Note that $\mathbf{e}_0$ and $\mathbf{e}_1$ form two edges of a thick rhomb, and $\mathbf{e}_0$ and $\mathbf{e}_2$ form two edges of a thin rhomb. Note also that $|\mathbf{e}_0 + \mathbf{e}_1 + \mathbf{e}_2| = \phi$.

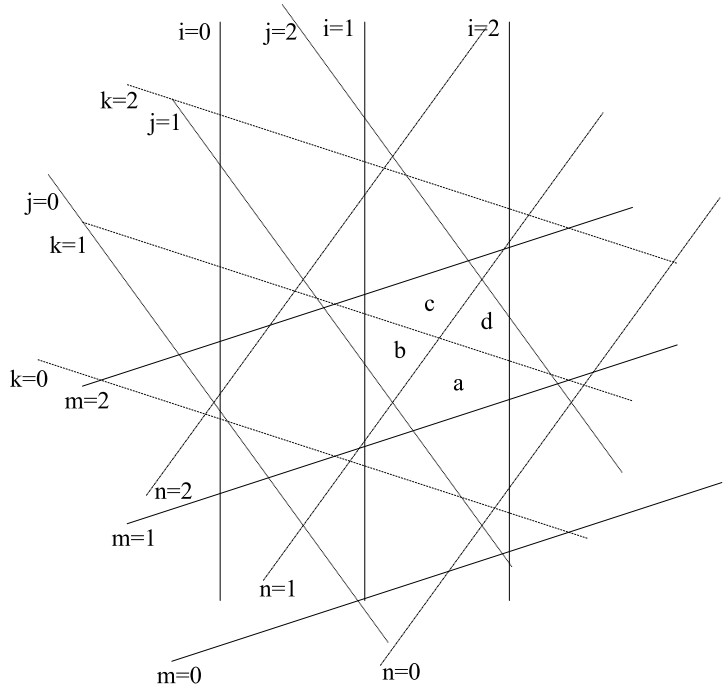

Fig. 11. A regular pentagrid. Each polygon formed by the grid lines is represented by the 5-tuple of grid labels (i, j, k, m, n): $a = (2, 2, 1, 2, 1)$; $b = (2, 2, 1, 2, 2)$; $c = (2, 2, 2, 2, 2)$; $d = (2, 2, 1, 2, 2)$. Note that adjacent polygons differ in only one label value (corresponding to the grid containing the grid line forming their common edge).

A *pentagrid* is a set of five grids, defined by five equally spaced direction vectors (Fig. 10b), and five offsets $\gamma_0, \ldots, \gamma_4$. de Bruijn [5] defines a *regular pentagrid* to be one where at most two grid lines cross at any one point (Fig. 11), achieved by suitable choice of offsets. (A sufficient condition for regularity is where all the offsets $-1 < \gamma_i < 1$, all are different, $\gamma_0 = 0$, and neither $\gamma_1 + \gamma_4$ nor $\gamma_2 + \gamma_3$ is an integer.)

Then, under the condition that

$$\sum_{i=0}^{4} \gamma_i = 0 \tag{3}$$

de Bruijn [5] proves that the pentagrid is the dual of a Penrose rhombus tiling: a tile is defined at every intersection of grid lines, and a tile vertex at every open polygonal region surrounded by grid lines; the condition that no more than two lines intersect ensures a unique tiling. ([21] extends some of these definitions and results to grids with irregular spacings of grid lines, including Ammann bar grids, and to 3D grids.)

Consider a pentagrid with each of its open regions labelled with a 5-tuple of integers $(k_0, k_1, k_2, k_3, k_4)$, calculated according to equation 2 (see Fig. 11). Each such region corresponds to the vertex of a rhomb, with coordinates [5, eqn 5.2] [21, eqn 4(−1)]

$$\mathbf{v} = \sum_{i=0}^{4} k_i \mathbf{e}_i \tag{4}$$

de Bruijn [5, eqn 5.4] defines

$$I = \sum_{i=0}^{4} k_i \bmod 5 \tag{5}$$

to be the *index* of the rhomb vertex, and shows that it is never 0.

The regularity constraint ensures that precisely four regions surround any intersection of pentagrid lines. These four regions differ in only two of their five grid labels (corresponding to grids containing the two intersecting grid lines they surround), and the values of these labels differ by at most one (Fig. 11). From equation 4 and Fig. 10b, we can therefore see that any such four adjacent regions correspond to the vertices of a thick rhomb (if the grid lines intersect at an angle of $2\pi/5$) or a thin rhomb (if the grid lines intersect at an angle of $4\pi/5$).

The fact that the rhombs so defined form a valid Penrose tiling is the basis of the pentagrid tiling algorithm.

3.6 Pentagrid lazy tiler algorithm

The pentagrid approach satisfies our requirements for a lazy tiler.

Requirement 1: Given a rectangular region to be tiled, T, we can lay a pentagrid over it, and extend the lines of the pentagrid to calculate all intersections that lie within T.

Requirement 2: Given a point $p \notin T$, we can extend the region to be tiled to T', the smallest rectangular region that includes both T and p, and then extend the pentagrid to calculate the new intersections that lie within the extended region $T' - T$.

This also satisfies requirement 3, that the tiling is not unnecessarily extended during this process.

Requirement 4: The algorithm calculates the vertices of rhombs before it calculates the rhombs themselves. These vertices are also vertices of neighbouring rhombs, so the neighbourhood can be built up as the tiles are calculated.

Requirement 5 can be satisfied as follows. Consider some further region definitions. The *neighbourhood* of a tile t, $N(t)$, is the set of all tiles close to t in some way (different kinds of Penrose tiling neighbourhoods suitable for defining CAs are given in section 4.1). Let C be the 'complete neighbourhood' region: all tiles t in T whose neighbourhood is 'complete' (that is, also in T) for some neighbourhood definition $N(t)$.

$$C = \{\ t \in T \mid N(t) \subseteq T\ \} \tag{6}$$

Clearly $C \subseteq T$, and unless the neighbourhood is trivial, $C \subset T$. Let A be the region where we allow cellular automaton activity to exist. We require

$$A \subseteq C \subset T \tag{7}$$

The extension of A is dictated by development of cellular automaton activity. This can be triggered when there is activity, a change in state, of a tile $t \in A$ where $N(t) \not\subseteq A$, so there are members of the neighbourhood of t outside A (although they will be in T if the requirement of equation 7 holds before this activity step). We extend A to include all the neighbourhood of t. In order to ensure that $A \subseteq C$ still holds, this may require extension of C, and hence of T.

It would be possible to calculate this extension exactly; however it is far cheaper in terms of computational complexity to extend the tiling such that the distance between the boundary of T and A is always greater than a distance d that ensures that the condition $A \subseteq C$ holds. (This does slightly violate requirement 2, that the extension not go beyond p, but not to any significant degree.)

For a generalised Moore neighbourhood (suitable for Game of Life CA rules) the value of d is determined by examining the valid vertex configurations in Figs. 4 and 5. For kites and darts, the largest is the queen (Q) vertex (Fig. 4), with a maximum diameter of twice the width of a dart, or $4\sin(2\pi/5)$ times the length of a short edge. For rhombs, the largest is the S3 vertex (Fig. 5), with a maximum diameter of twice the long diagonal of the thin rhomb, or $4\sin(2\pi/5)$ times the length of an edge.

The original treatment [6] uses the pentagrid approach to produce a rhomb tiling; here we outline how it is used in an algorithm for a lazily extending tiling, of rhombs, and of kites and darts.

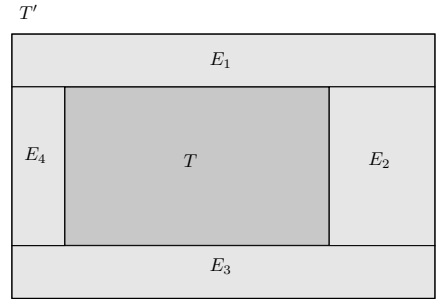

Fig. 12. Initial tiling T extends to tiling T' with extension regions E_1, E_2, E_3, E_4

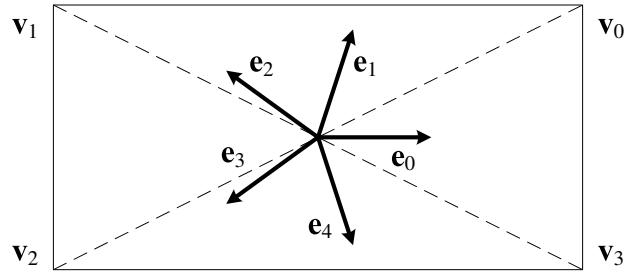

Fig. 13. An arbitrary rectangle R to be tiled, with corners $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$, and diagonals shown with dashed lines. For each $\mathbf{e}_i$, the diagonal with direction closer to that of $\mathbf{e}_i$ gives the two opposite corners between which grid lines should be calculated. So $\mathbf{v}_0$ and $\mathbf{v}_2$ are used to define the extent of grid lines for $\mathbf{e}_1$ and $\mathbf{e}_3$; $\mathbf{v}_1$ and $\mathbf{v}_3$ for $\mathbf{e}_2$ and $\mathbf{e}_4$ either pairs of opposite corners may be used for $\mathbf{e}_0$.

Choosing the offsets. First, choose suitable offsets γ_i that ensure a regular pentagrid and that obey equation 3. A sufficient such choice is $(0, \gamma, -\gamma, 2\gamma, -2\gamma)$, governed by the single parameter γ, where $0 < \gamma < 0.5$.

In all the results reported here, we used the offsets $(0.2, -0.2, -0.1, 0.4, -0.3)$. This violates the $\gamma_0 = 0$ condition, but does create regular pentagrid.

Tiling a region. We consider only the tiling of a rectangular region. The problem of tiling an initial region and then extending the tiling can be reduced to tiling arbitrary rectangles. For example, Fig. 12 shows the extension of tiling T to tiling T' with four rectangular extension regions E_1, E_2, E_3, E_4.

To tile an arbitrary rectangle R we must ensure that all grid intersections that lie within the rectangle are calculated. For each of the five grids, every one of its lines that intersects the rectangle must be considered. We do this by observing that the maximum extent of the grid that needs to be considered is defined by the rectangle's diagonal that is closer to the direction of the grid, $\mathbf{e}$ (Fig. 13). So, for example, grid 1 has the set of grid values N_1 given by (using

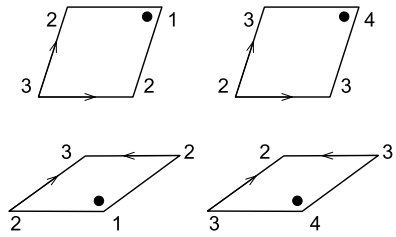

Fig. 14. Marking a rhombus depending on its vertex indices. These are the only combinations of vertex indices that are generated from a pentagrid. (Note: these vertex index numbers are unrelated to the 'vertex angle numbers' shown in Fig. 3a.)

equation 2)

$$\lceil \mathbf{v}_2.\mathbf{e}_1 - \gamma_1 \rceil < N_1 < \lceil \mathbf{v}_0.\mathbf{e}_1 - \gamma_1 \rceil \tag{8}$$

All pairs drawn from $N_0 \times N_1 \times N_2 \times N_3 \times N_4$ calculated in this way contain all the grid intersections in the rectangle R (plus more outside).

Converting grid intersections to tile vertices. To calculate the vertices of a tile we must discover the 5-tuples defining the four open regions around an intersection. The 5-tuples corresponding to each intersection index are calculated using equation 2 (to get the other three indices of the intersection 5-tuple) and Fig. 11 (to get the four vertex 5-tuples). The coordinates of the corresponding vertex are given by equation 4.

Iterating over intersections naively would result in each vertex being calculated 4 times: to ensure the each vertex is calculated only once, we store the vertex information in a hashtable indexed by its 5-tuple. At each vertex we also maintain a list of all tiles that share the vertex: this aids subsequent neighbourhood calculations.

Conversion from rhomb to kite and dart tiling. The pentagrid approach yields a rhomb tiling. This can subsequently be converted to a kite and dart tiling using the mapping shown in Fig 7.

First, the rhombs need to be "marked", in a way that depends on the vertex index (equation 5). de Bruijn [5] shows that the only combinations of vertex indices formed by a pentagrid are $(1, 2, 2, 3)$ or $(2, 3, 3, 4)$; the marked vertex is the one with vertex index 1 or 4 (Fig. 14).

Each thick rhomb is broken into a dart and two half-kites, each thin rhomb is broken into two half-kites (Fig. 7), taking the rhomb marking into account. The half-kites are then suitably joined to produce a complete kite and dart tiling.

Precision issues. We have the condition that only two grid lines cross at any intersection point. de Bruijn [7] gives the condition for a singular pentagrid

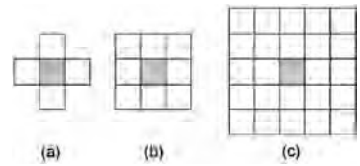

Fig. 15. (a) von Neumann neighbourhood (b) Moore neighbourhood, or box neighbourhood of radius $r = 1$ (c) box neighbourhood of radius $r = 2$

(where more than two lines intersect at some point) to be for grid $i \in \{0, 1, 2, 3, 4\}$ and an integer k if

$$(k - \gamma_i)\phi + \gamma_{i-1} + \gamma_{i+1} \in \mathbb{Z} \tag{9}$$

Choosing grid $1 = 0$, $\gamma_0 = 0$, this gives

$$k\phi + \gamma_1 + \gamma_4 \in \mathbb{Z} \tag{10}$$

Assuming non-extreme choices of γ_1 and γ_4 (that is, choosing $\gamma_1 + \gamma_4 = O(1)$), and given $\phi = O(1)$, this condition could occur in a computation by rounding error, if there were an error of one unit in the last place of k.

Our implementation uses Java doubles, which are implemented using double-precision 64-bit IEEE 754 floating point, which have a 53-bit mantissa, corresponding to approximately 16 decimal digits. So the algorithm does not lose accuracy (no grid singularity occurs) below $N = O(10^{16})$ grid lines, or $O(10^{32})$ tiles. If larger tilings than this are required, higher precision arithmetic should be used. (This is only an order of magnitude argument, but since our tiling investigations are significantly below this limit, with $O(10^5)$ tiles, we can be confident that there is no loss of accuracy affecting our results.)

4 Cellular automata on aperiodic lattices

Classic cellular automata are defined on regular lattices. The update rule depends on the state of each cell's full neighbourhood (the surrounding cells, and the updating cell itself)[3], and the structure of that neighbourhood is invariant: all places in the lattice look the same, and the update rule can be applied uniformly across the lattice. Typical neighbourhoods for 2D cellular automata are shown in Fig. 15. These neighbourhoods can be formally defined in terms of metrics on the lattice. However, we define them (later) in an equivalent manner that permits easy generalisation to aperiodic lattices.

[3] The standard definition of CA neighbourhood includes both the surrounding cells *and* the updating cell. Throughout this paper we use slightly different terminology (because we are nearly always referring to outer totalistic CA rules, see later): by *neighbourhood* we mean only the surrounding cells. If we want to include the updating cell, we refer to the *full neighbourhood*.

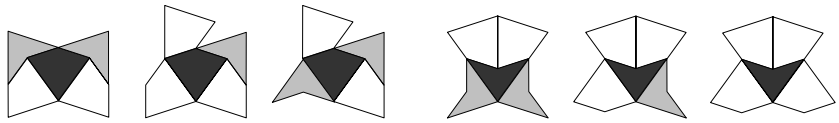

Fig. 16. The generalised von Neumann neighbourhoods of a kite and dart Penrose tiling.

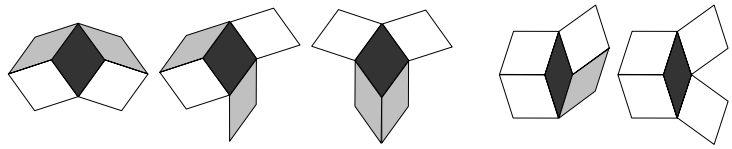

Fig. 17. The generalised von Neumann neighbourhoods of a rhomb Penrose tiling.

In general, the update rule depends on the particular state of each separate neighbour. For *totalistic* CA rules, however, the next state of a cell depends only on the number of full neighbourhood cells in certain states. For *outer totalistic* CA rules, the next state of a cell depends only on its current state, and the number of neighbourhood cells in certain states.

For example, in Conway's *Game of Life* outer totalistic CA, the neighbourhood of each cell comprises the 8 nearest cells of the regular Moore neighbourhood (Fig. 15b). Each cell has two states, 'dead' and 'alive'. If a cell is alive at time t, then it stays alive iff it has 2 or 3 live neighbours (otherwise it dies of 'loneliness' or 'overcrowding'). If a cell is dead at time t, then it becomes alive (is 'born') iff it has exactly 3 live neighbours.

For aperiodic lattices such as a Penrose tiling, the detailed structure of the neighbourhood varies at different locations in the lattice. Totalistic and outer totalistic rules can be given an interpretation in these aperiodic tiling neighbourhoods.

4.1 Generalised von Neumann neighbourhood

We define the generalised von Neumann neighbourhood of a cell to be all the cells with which it shares an edge (or, equivalently, two distinct vertices). Hence the size of the neighbourhood equals the number of edges of the central cell. Figures 16 and 17 show the range of distinct generalised von Neumann neighbourhoods which form valid vertices (rotations and mirror images of these neighbourhoods are not considered to be distinct).

de Bruijn [7] identifies the same rhomb neighbourhoods (but considers mirror images separately), and shows that a valid Penrose rhomb tiling can be constructed by considering just these neighbourhoods, without the need to use the rhomb matching rules of Fig. 3.

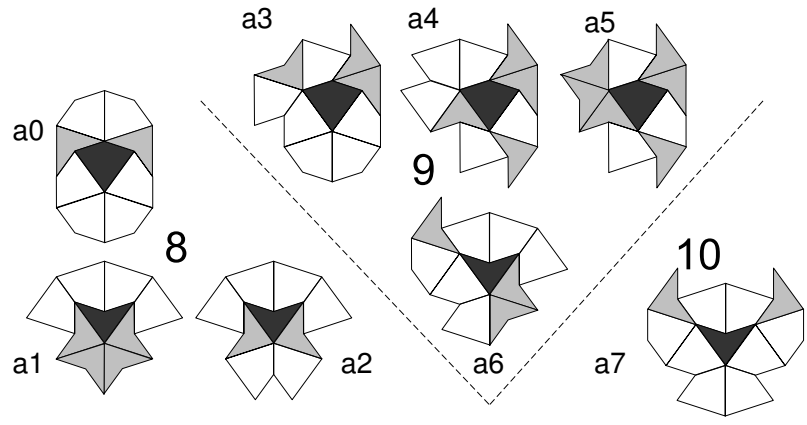

Fig. 18. The generalised Moore neighbourhoods on a kite and dart Penrose tiling, with neighbourhood sizes.

In the rectangular lattice (Fig. 15a), none of the four von Neumann neighbourhood cells themselves share an edge. So if A is a neighbour of B, and B is a neighbour of C, then A is *not* a neighbour of C: neighbouring von Neumann neighbourhoods do not overlap (recall that we do not treat the central site as a member of the neighbourhood for the purposes of this paper). In the Penrose lattice, this is no longer the case: cells in a generalised von Neumann neighbourhood can share an edge, so neighbouring generalised von Neumann neighbourhoods can overlap. This may affect the communication paths through the Penrose CA.

4.2 Generalised Moore neighbourhood

We define the generalised Moore neighbourhood of a cell to be all the cells with which it share a vertex.

Not only do cells have irregular shaped neighbourhoods, with the generalised Moore neighbourhood not all cells have the same number of neighbours. The range of neighbourhood sizes and configurations is limited.

Fig. 18 shows the eight distinct generalised Moore neighbourhoods in a kite and dart tiling: there are no other valid ways to surround a kite or a dart (this can be established by exhaustive consideration of the valid vertex configurations shown in Fig. 4). So there is one neighbourhood configuration of size 8 around a kite, and two around a dart; three of size 9 around a kite, and one around a dart; and one of size 10, around a dart. ([11] incorrectly states that kite and dart tilings have neighbourhoods of size 8 and 9 only.) Figure reffigure:moorekitenbrhood show an area of kite and dart tilings with colouring to highlight the size of cells' neighbourhoods.

Similarly, Fig. 20 shows the 11 distinct generalised Moore neighbourhoods in a rhomb tiling. There is a larger range of distinct neighbourhood configurations

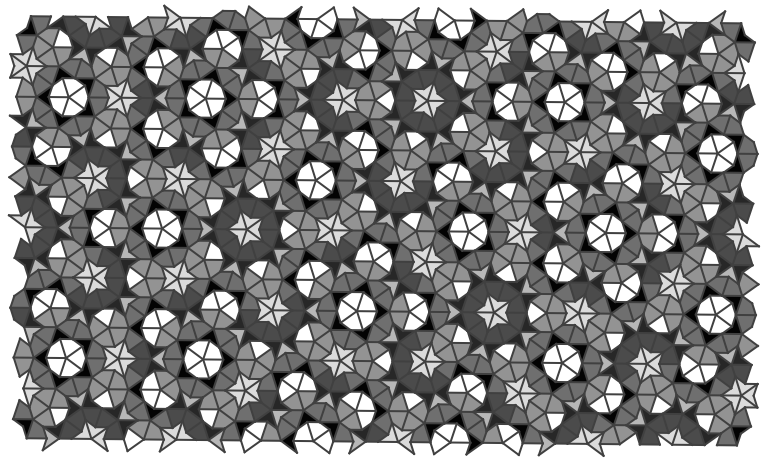

Fig. 19. A kite and dart tiling shaded by neighbourhood type. The neighbourhood shading is uniformly distributed between white and black such that a_0 is white and a_7 black.

for rhomb tilings. Figure 19 show an area of rhomb tilings with colouring to highlight the size of cells' neighbourhoods.

As can be seen from Figs. 19 and 21, not all sizes of neighbourhoods appear with the same frequency. Figure 22 shows the distribution of neighbourhood sizes in a kite and dart tiling and in a rhomb tiling.

4.3 Generalised box neighbourhood

The Moore neighbourhood is a special case of a box neighbourhood, with radius $r = 1$. Let $N(c)$ be the generalised Moore neighbourhood of cell c. We define recursively a generalised box neighbourhood of radius $r > 1$. Let $N(c, r)$ be the box neighbourhood of cell c, of radius r. Then define

$$N(c, 1) = N(c) \tag{11}$$

$$N(c, r) = \bigcup_{n \in N(c)} N(n, r - 1) \tag{12}$$

See Figs. 23 and 24.

The frequency distribution of neighbourhood sizes for $r = 2$ neighbourhoods is shown in Fig. 25 for kites and darts, and for rhombs. Again, the rhomb tilings have larger neighbourhoods, and a larger spread of neighbourhood sizes.

4.4 Penrose Life rules

Using our definition of the generalised Moore neighbourhood, the definition of the Game of Life as given in section 4 can be used unchanged on a Penrose

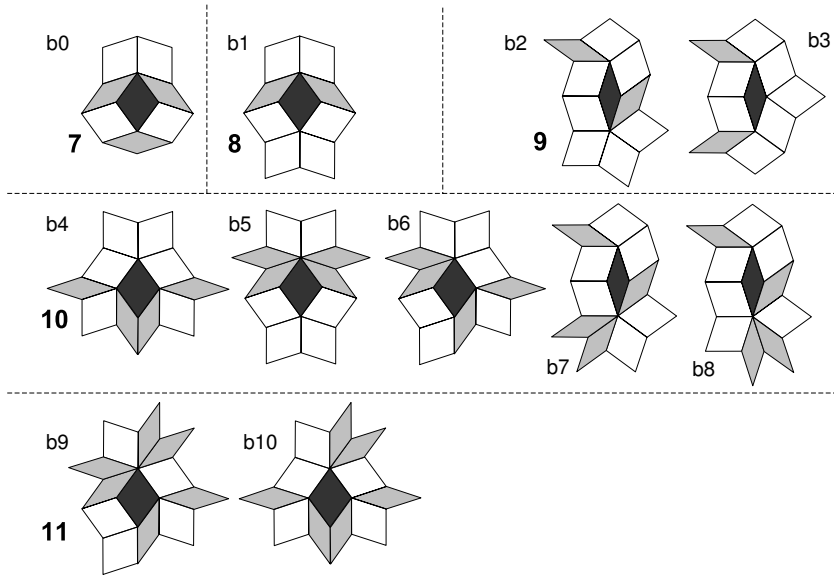

Fig. 20. The generalised Moore neighbourhoods on a rhomb Penrose tiling, with neighbourhood sizes.

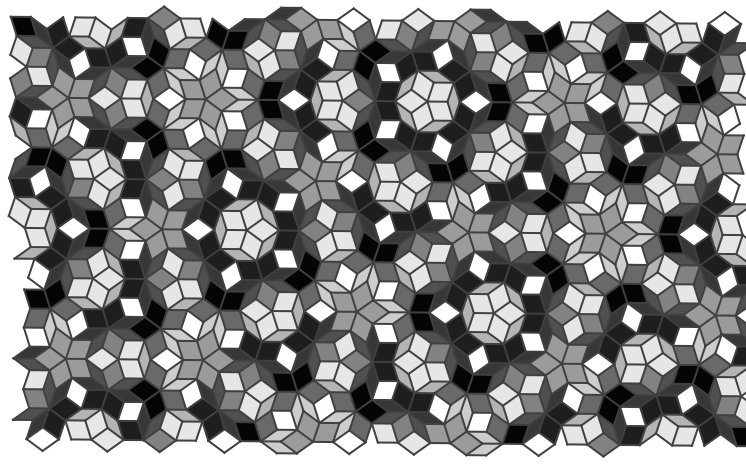

Fig. 21. A rhomb tiling shaded by neighbourhood type. The neighbourhood shading is uniformly distributed between white and black such that b_0 is white and b_{10} black.

	kite/dart			rhomb		
size	type	cells	%	type	cells	%
7				b0	2831	9.1
					2831	9.1
8	a0	4994	14.7	b1	4576	14.6
	a1	4248	12.5			
	a2	1890	5.6			
		11132	32.7		4576	14.6
9	a3	6116	18.0	b2	2134	6.8
	a4	6125	18.0	b3	2842	9.1
	a5	3762	11.1			
	a6	3774	11.1			
		19777	58.2		4976	15.9
10	a7	3083	9.1	b4	2370	7.6
				b5	1735	5.6
				b6	2133	6.8
				b7	3475	11.1
				b8	3501	11.2
		3083	9.1		13214	42.3
11				b9	3522	11.3
				b10	2136	6.8
					5658	18.1

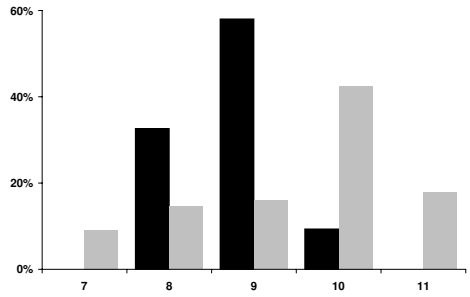

Fig. 22. Generalised Moore neighbourhood statistics, on a 33992 cell kite and dart tiling (black bars, median size = 9), and a 31255 cell rhomb tiling (grey bars, median size = 10)

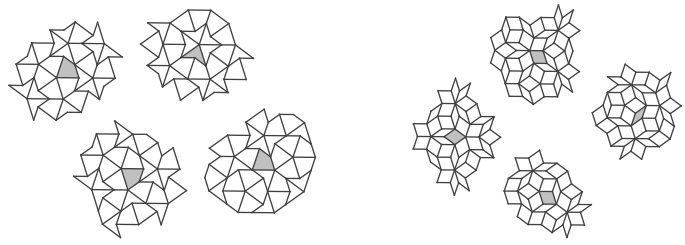

Fig. 23. Examples of box radius $r = 2$ aperiodic neighbourhoods on kite and dart Penrose tilings, and on rhomb Penrose tilings

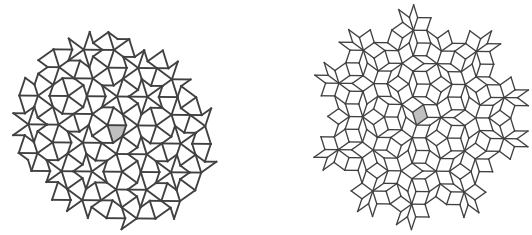

Fig. 24. An example of a box radius $r = 5$ aperiodic neighbourhood on a kite and dart Penrose tiling, and on a rhomb Penrose tiling

size	kite/dart		rhomb	
	cells	%	cells	%
25	4248	12.5		
26	4994	14.7	1597	5.1
27	4204	12.4	829	2.7
28	11330	33.3	1912	6.1
29	2351	6.9	1092	3.5
30	3782	11.1	1977	6.3
31	1441	4.2	1314	4.2
32	1642	4.8	2588	8.3
33			493	1.6
34			4609	14.7
35			1075	3.4
36			4622	14.8
37			3489	11.2
38			828	2.6
39			4830	15.5

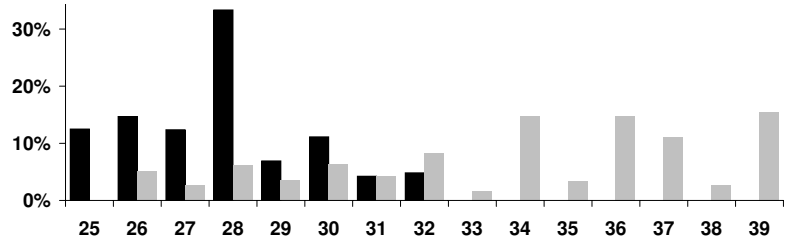

Fig. 25. Neighbourhood statistics $r = 2$ neighbourhood, on a 33992 cell kite and dart tiling (black bars, median size = 28), and a 31255 cell rhomb tiling (grey bars, median size = 34)

lattice. Some early investigations are reported in [11]; further investigations are reported later in this paper.

In our investigations, we use some typical GoL terminology, defined here. (The quoted definitions are from [19].)

soup "*A random initial pattern, often assumed to cover the whole Life universe.*" Here we consider only finite soup extents, but allow subsequent activity outside the initial soup patch.

quiescence Eventual periodic CA activity. Once the CA has entered a quiescent state, its future activity is periodic, and hence predictable.

ash "*The (stable or oscillating) debris left by a random reaction.*" Hence an ash is the quiescent state left by a soup.

5 Experimenting with Life

In [11] we report that the Game of Life has different quantitative behaviour on a regular lattice and on a Penrose kite and dart lattice: on the Penrose lattice the lifetime to quiescence is much shorter, and the ash density is lower. This section investigates if there are similar differences between the behaviour of the rules running on kite and dart and on rhomb lattices.

Null Hypothesis: The Game of Life run on kites and darts has identical statistical behaviour to the Game of Life run on rhombs.

To test this hypothesis, we investigate three statistics: lifetime to quiescence, ash density, and growth of the active area.

5.1 Experimental setup

To test the hypothesis we vary the density D of soups of similar sizes S on rhomb and kite and dart tilings, run the cellular automaton to quiescence, and record the lifetime to quiescence t_q, ash density ρ (measured over the soup box), and soup growth g.

Lifetime t_q: The lifetime, or the time to quiescence, is defined to be the number of timesteps from the soup state ($t = 1$) until the pattern of live cells (measured over the whole timing G) first repeats (at $t = t_q$). Each timestep, the CA's current state is stored, along with the number of live cells. To check for quiescence, the current state is compared to all previous states with the same number of live cells. The period p is the number of timesteps since the state was previously seen: $p = t_q - t_{prev}$.

Ash density ρ: The proportion of live cells in the ash at $t = t_q$, measured over the soup tiles.

Soup growth g: The number of cells in the maximum active area divided by the number of cells in the soup: $g = A/S$. measured over the soup tiles (Fig. 26).

Tiling grid: We use a lazily expanding tiling for both kites and darts, and rhombs. We use an initial tiling of size $G = 23194$ for the kite and dart experiments, and of size $G = 23123$ for the rhomb experiments. It is difficult to produce

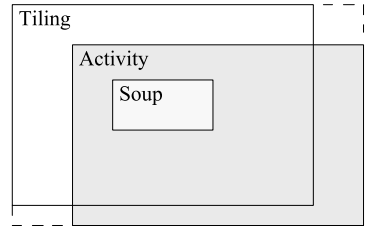

Fig. 26. The initial tiling grid G, the central soup area S, the maximum activity area during the run A, and the possibly extended tiling grid G_q (dashed box) to accomodate the activity.

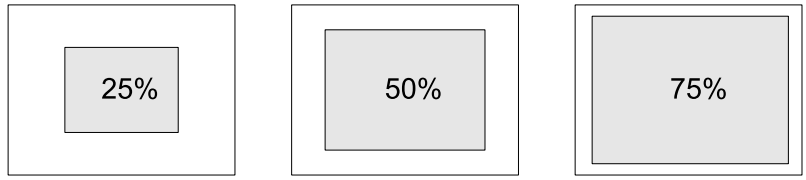

Fig. 27. The three central soup areas, to scale with the initial grid area.

identical sized tilings: these are deemed close enough for fair comparison. These differences in tile numbers are of similar scale to the differences in tile numbers between regular and kite and dart tilings used in [11] (and are about twice the size of the largest grid explored there).

Soup area: Three initial soup areas S, covering the central 25%, 50%, 75% of the area of the tiling. See Fig. 27 and 28.

Soup density: 100 soup densities D, in the range $[0.01, 1.0]$ with increments of 0.01. Each cell in the soup area S is initially alive with probability D; all other cells in G are initially dead. See Fig. 29.

Runs: Each of the 100 soup densities D across the three soup sizes S is run to quiescence 1000 times.

5.2 Statistical analysis

We want to test whether certain distributions are statistically the same or different: the commonly-used tests assume an underlying normal distribution. Are the distributions here (sufficiently) normal?

Figures 31 and 32 show the histograms of lifetime and ash density results over the 1000 runs for one particular soup size and soup density. The lifetime distributions, at least, do not look normal.

We investigate further the distribution of lifetimes and ash densities for these examples. We calculate the median, mean, standard deviation, skew and kurtosis of these distributions (using the MS-Excel functions MEDIAN, AVERAGE, STDEV, SKEW, and KURT respectively), for the lifetimes (Fig. 33) and the ash densities

	25%	50%	75%	G
kite and dart	5842	11670	17527	23194
rhomb	5815	11611	17405	23123

Fig. 28. Number of tiles involved in the experiments, soup sizes $S = 25\%$, 50% and 75%, and full initial grid size G

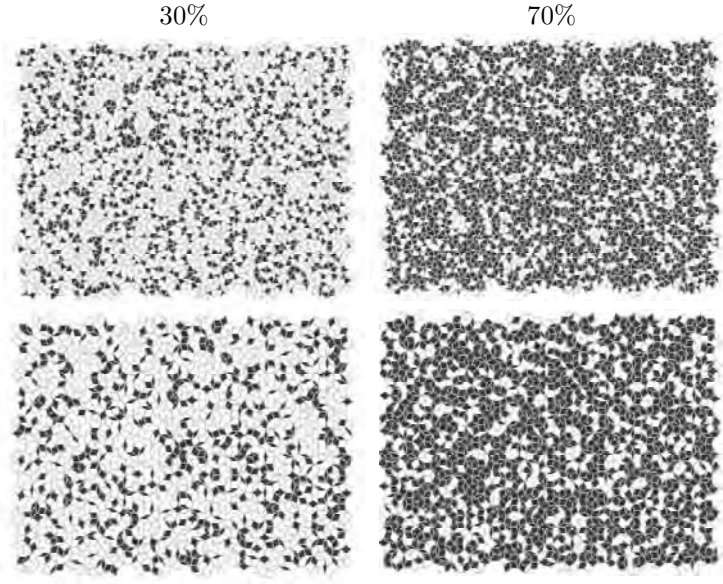

Fig. 29. Typical soups at two densities, for kite and dart (top) and rhomb (bottom) tilings

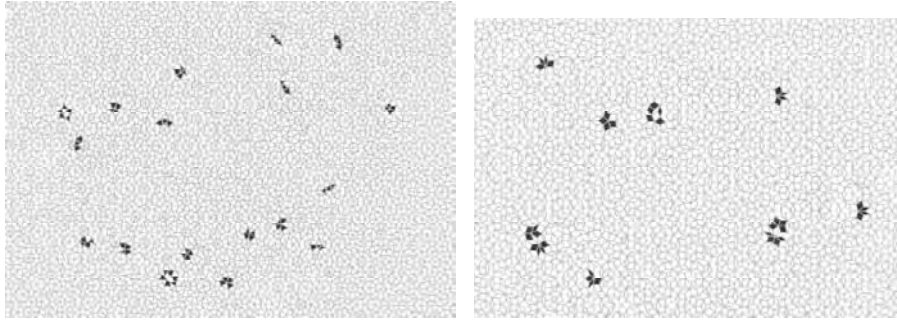

Fig. 30. The ashes resulting from the 30% soups of Fig. 29. Note the extended areas of activity.

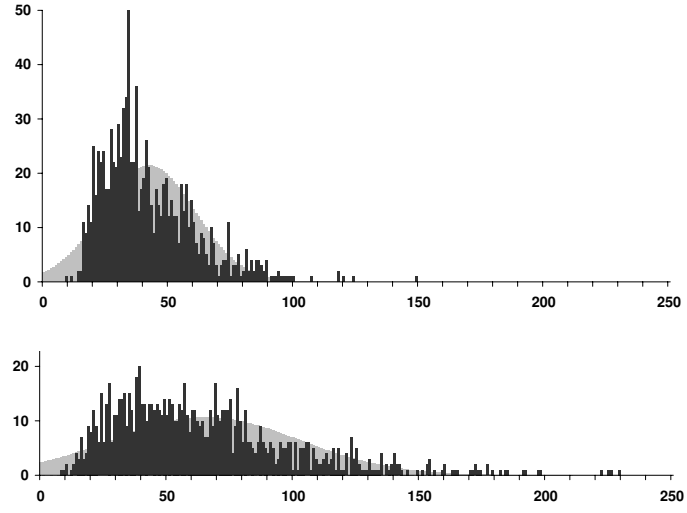

Fig. 31. The distribution of lifetimes to quiescence on the kite and dart tiling (top) and rhomb tiling (bottom), for 1000 runs with soup size $S = 25\%$ and soup density $D = 0.8$; with comparison normal distributions of the same mean and standard deviation.

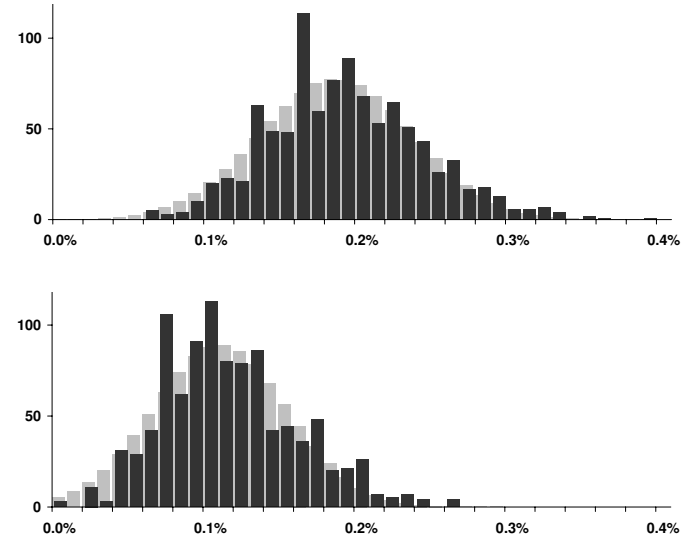

Fig. 32. The distribution of ash densities on the kite and dart tiling (top) and rhomb tiling (bottom), for 1000 runs with soup size $S = 25\%$ and soup density $D = 0.8$; with comparison normal distributions of the same mean and standard deviation.

		$D = 0.4$		$D = 0.8$	
soup		k&d	rhomb	k&d	rhomb
25%	m	96	158	37	57.5
	μ	99.4	163.0	41.9	65.1
	σ	19.6	37.0	18.6	37.3
	s	1.1	0.9	1.2	1.5
	k	2.0	1.2	2.1	3.9
50%	m	108	179	40	60
	μ	111	185.1	44.7	66.6
	σ	19.6	37.2	18.5	33.7
	s	0.7	0.8	1.2	1.2
	k	0.9	0.5	2.0	2.1
75%	m	116	190	44	67
	μ	118.6	198.1	47.1	74.1
	σ	20.0	40.4	17.9	35.9
	s	0.9	1.2	1.3	1.2
	k	1.1	2.8	4.3	4.3

Fig. 33. Statistics for the lifetime distributions (median m, mean μ, standard deviation σ, skew s, kurtosis k) for soup densities $D = 0.4$ and 0.8; soup sizes $S = 25\%$, 50% and 75%

		$D = 0.4$		$D = 0.8$	
soup		k&d	rhomb	k&d	rhomb
25%	μ	0.0044	0.0034	0.0018	0.0011
	σ	0.0008	0.0008	0.0005	0.0004
	s	0.2	0.3	0.4	0.5
	k	0.2	−0.0	0.2	0.1
50%	μ	0.0084	0.0022	0.0063	0.0015
	σ	0.0011	0.0006	0.0010	0.0005
	s	−0.1	0.2	0.1	0.3
	k	0.1	−0.1	−0.0	0.0
75%	μ	0.0123	0.0091	0.0029	0.0023
	σ	0.0027	0.0013	0.0007	0.0006
	s	−0.1	0.2	0.2	0.1
	k	−0.1	−0.1	0.0	−0.1

Fig. 34. Statistics for the ash densities (mean μ, standard deviation σ, skew s, kurtosis k) for soup densities $D = 0.4$ and 0.8; soup sizes $S = 25\%$, 50% and 75%

	kites	rhombs
$\# > 45$	349 ($= A$)	650 ($= B$)
$\# \leq 45$	651 ($= C$)	350 ($= D$)

Fig. 35. The number of measurements above, and not above, the joint median value of 45, for soup size $S = 25\%$, density $D = 0.8$

(Fig. 34: we do not show medians here, since they are indistinguishable from the means).

For large samples ($N > 150$) drawn from a normal population, the skewness statistic is approximately normally distributed with mean 0 and standard deviation $s_s = \sqrt{6/N}$ [20, §5.13]; for very large samples ($N > 1000$) drawn from a normal population, the kurtosis statistic is approximately normally distributed with mean 0 and standard deviation $s_k = \sqrt{24/N}$ [20, §5.14]. Hence skew values beyond two standard errors of skewness, or kurtosis values beyond two standard errors of kurtosis, indicate that the distribution is not normal at the 95% confidence level.

For $N = 1000$ (just valid for the kurtosis test), $2s_s = 0.5$ and $2s_k = 1.0$. Both these values are lower than those calculated for the lifetimes (Fig. 33), so the lifetime distributions are not normal at the 95% confidence level. Normality of the ash densities has not been ruled out by this test (Fig. 34).

Given this non-normality of the lifetimes, we calculate the non-parametric median and quartile statistics of the runs, for the range of soup densities (Figs. 37 and 39). These results are in qualitative agreement with those in [11]: low lifetimes and ash densities at extreme soup densities; a 'plateau' in the behaviours for soup densities $\sim 0.2 - 0.6$; lifetimes $\sim 100 - 200$; ash densities $\sim 1 - 2\%$. We now, however, have better statistics, and new results for rhomb tilings.

Since the lifetime distributions are not normal, we use the non-parametric *median test*, to test whether the distributions have statistically significantly different medians [18, pp.111–115]. (In practice, our sample sizes are probably large enough that assuming normality and using a t-test is probably valid. However, the certainly valid, if somewhat weaker, non-parametric test is adequate in this case.)

Null Hypothesis T: for soup size $S = 25\%$, density $D = 0.8$, there is no difference between the median lifetimes for kites and darts, and for rhombs.

The calculation involves splitting the measurements into four groups: those above, and not above, the joint median of the measurements. The relevant numbers for our test case are given in Fig. 35). Then we calculate the value of χ^2 from [18, eqn(6.4)]:

$$\chi^2 = \frac{N\left(|AD - BC| - N/2\right)^2}{(A+B)(C+D)(A+C)(B+D)} = 90.3 \tag{13}$$

The probability of occurrence under Null Hypothesis T for $\chi^2 \geq 90.3$ with one degree of freedom is $p < \frac{1}{2}\,\text{CHIDIST}(90.3,1) = 10^{-21}$ for a one-tailed test.

Therefore we can reject Null Hypothesis T, with an *extremely* high degree of statistical confidence.

In fact, the difference in the medians in this test case is statistically significant to an almost ludicrous degree. This extreme level of statistical confidence is due mostly to the large number of samples, $N = 1000$. (Such large samples are much more typical in computer science than, say, medicine, because computer experiments are relatively cheap, and have no ethical considerations.) As Bakan says ([2, ch.1, p.7], as quoted in [12]): "there is really no good reason to expect the null hypothesis to be true in any population". A sufficiently large sample size will always be able to refute a null hypothesis: the smaller the effect, the larger the sample required to detect it. For normally-distributed populations with means and standard deviations similar to those of Fig. 34, sample sizes in the low tens would be sufficient to establish a statistically significant difference of their means at the 99% confidence level.

Because of this, we also perform a test of the *effect size*. We use Cohen's effect size d-test [4, §2.5]. (Strictly speaking, we should not use this statistic, because the distributions are not normal. But if we get a sufficiently large value of the d-statistic, we can still be confident in the importance of the difference.) For samples with different variances but the same sample size, we use

$$d = \frac{m_1 - m_2}{\sqrt{(s_1^2 + s_2^2)/2}} \tag{14}$$

where the m_i are the two sample means, and the s_i are the two sample variances. So d measures the difference in the means compared to the spread of the data.

Cohen's criterion is that $d = 0.2$ indicates a small effect, $d = 0.5$ a medium effect, and $d = 0.8$ a large effect. For soup size $S = 25\%$, density $D = 0.8$, we have $d = 0.8$ indicating a large effect from the change in the tiling.

So, for all the results that follow, we do not present the statistical significance: the differences are all extremely significant. We present the skew and kurtosis normality tests, median and quartiles, means, and the effect size, demonstrating that all the statistics chosen exhibit a large effect with the change in the tiling.

6 Lifetime, ash, growth results

6.1 Lifetimes

Null Hypothesis 1: The Game of Life run on kites and darts has identical lifetime statistics to the Game of Life run on rhombs.

See Figs. 36, 37. The skew and kurtosis tests show that the distributions are significantly non-normal. The lifetime distributions for the two tilings are different, with a large effect size, refuting Null Hypothesis 1. The Game of Life on the rhomb tiling has significantly longer lifetimes than it does on the kite and dart tiling. From [11], we can say that they both have shorter lifetimes than Life on a regular lattice.

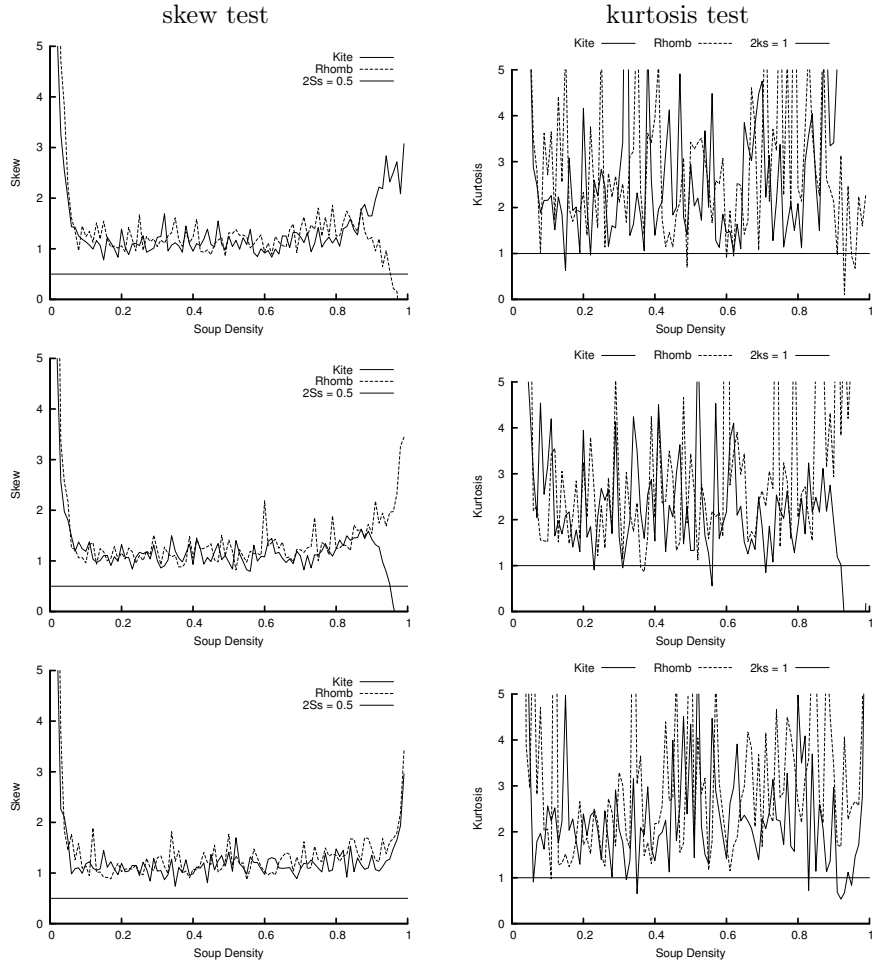

Fig. 36. Lifetime to quiescence t_q: normality tests (soup sizes 25% top, 50% middle, 75% bottom)

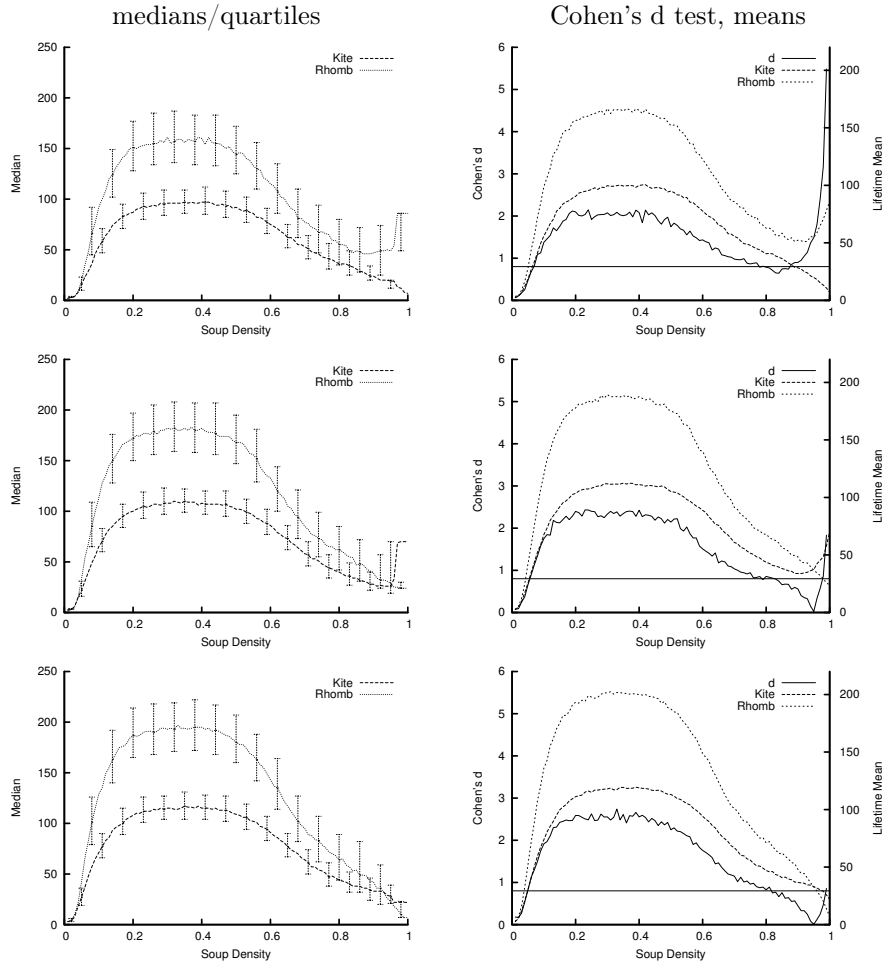

Fig. 37. Lifetime to quiescence t_q: medians and effect size (soup sizes 25% top, 50% middle, 75% bottom)

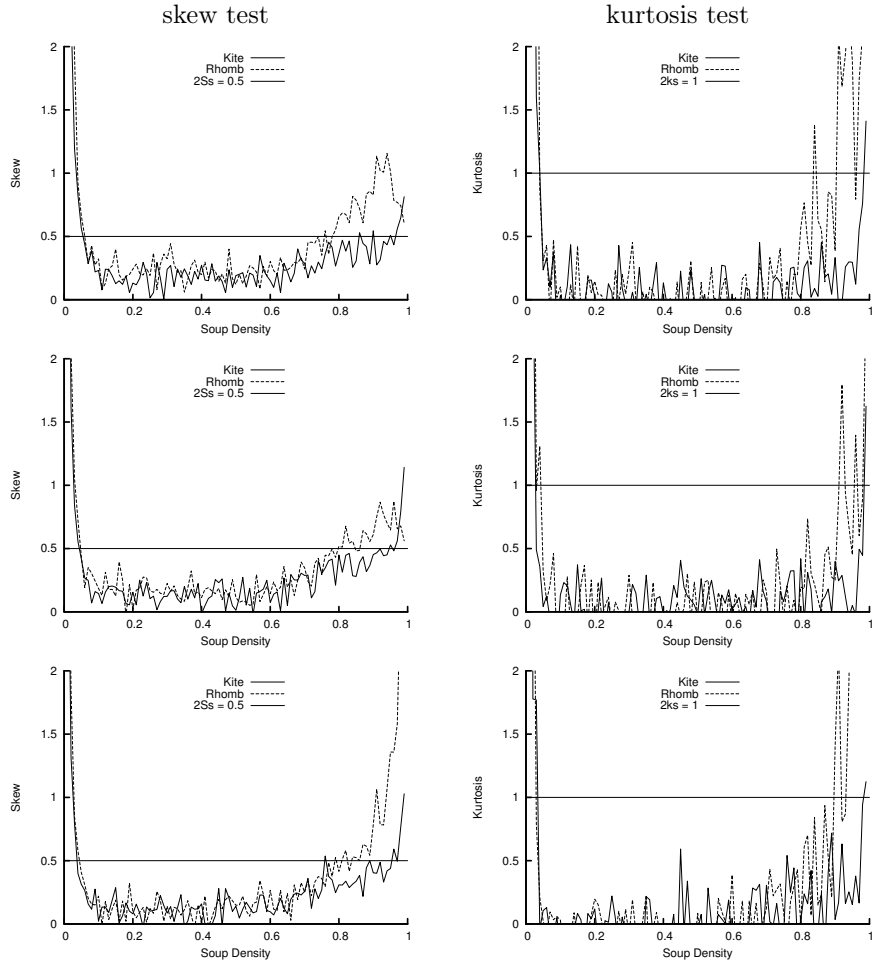

Fig. 38. Ash density ρ: normality tests (soup sizes 25% top, 50% middle, 75% bottom)

6.2 Ash densities

Null Hypothesis 2: The Game of Life run on kites and darts has identical ash density statistics to the Game of Life run on rhombs.

See Figs. 38, 39. The skew and kurtosis tests show that the distributions are consistent with being normal, except for large soup densities. The ash density distributions for the two tilings are different, with a large effect size, refuting Null Hypothesis 2. The Game of Life on the rhomb tiling has significantly lower ash densities than it does on the kite and dart tiling. From [11], we can say that they both have lower ash densities than Life on a regular lattice.

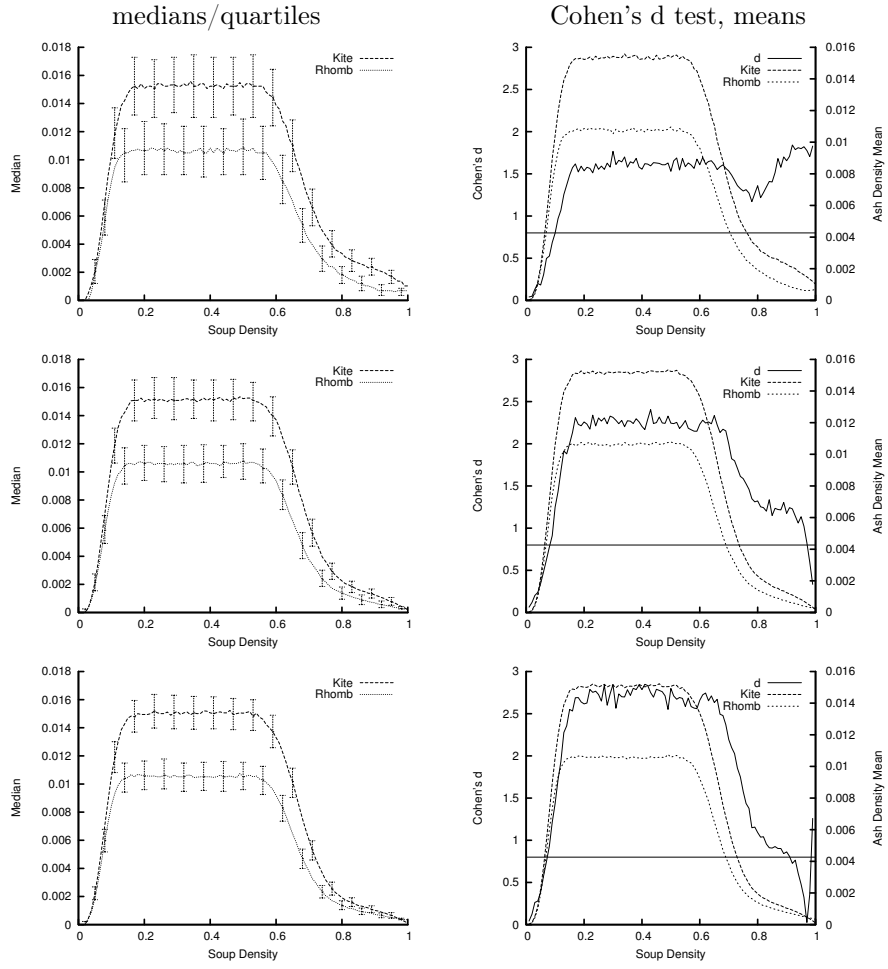

Fig. 39. Ash density ρ: medians and effect size (soup sizes 25% top, 50% middle, 75% bottom)

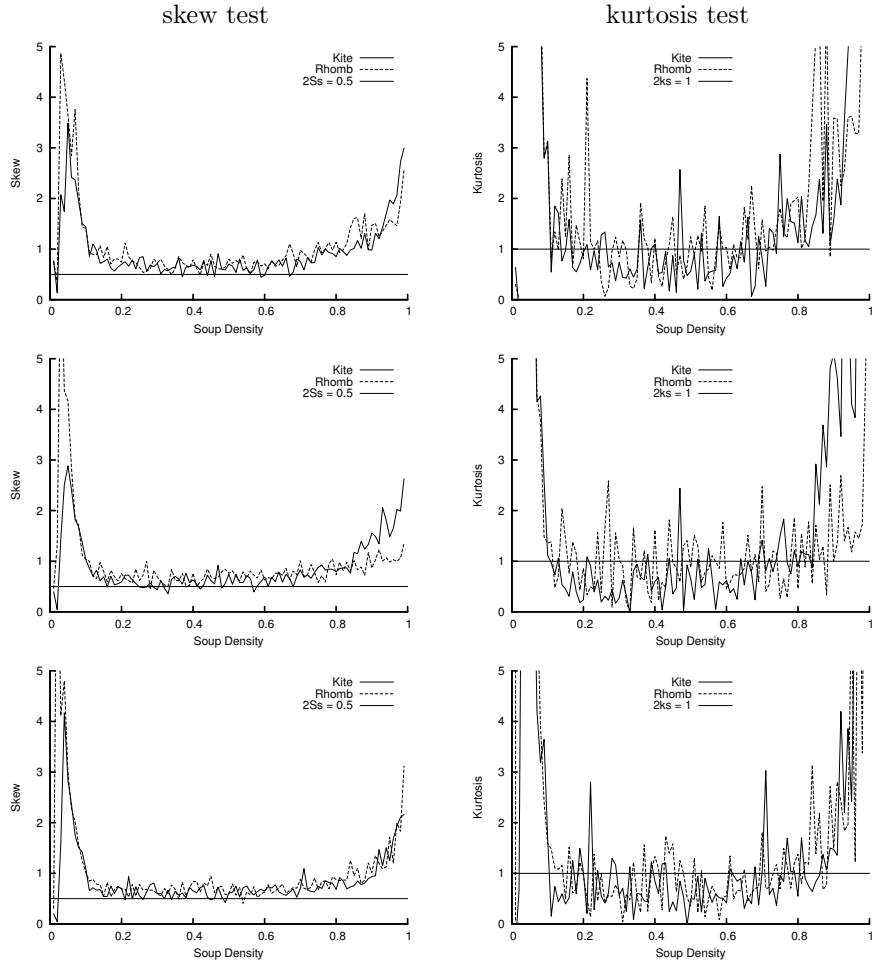

Fig. 40. Soup growth g: normality tests (soup sizes 25% top, 50% middle, 75% bottom)

6.3 Soup growth

Null Hypothesis 3: The Game of Life run on kites and darts has identical growth of soup to the Game of Life run on rhombs.

See Figs. 40, 41 for statistics on the growth of the area of soup. The skew and kurtosis tests show that the distributions are significantly non-normal. The growths of the two tilings are different, with a large effect size, refuting Null Hypothesis 3. The Game of Life on the rhomb tiling has significantly more growth from soup than it does on the kite and dart tiling.

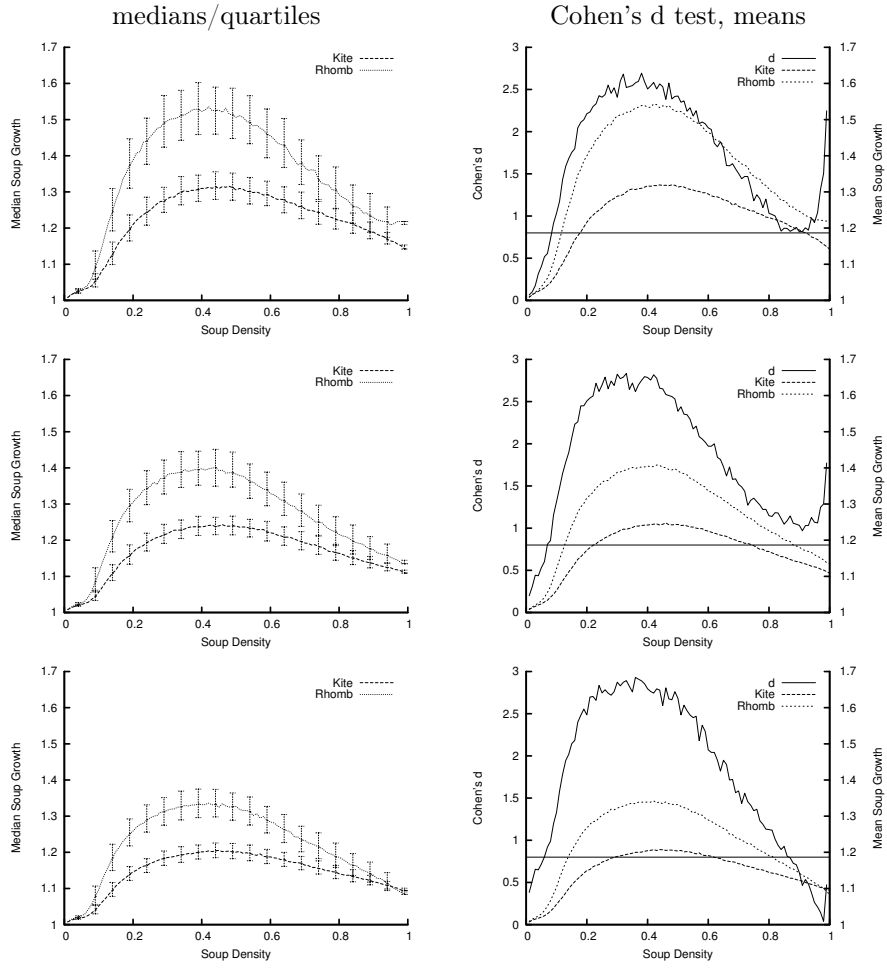

Fig. 41. Soup growth g: median and effect size (soup sizes 25% top, 50% middle, 75% bottom)

7 Conclusions

We have presented a Penrose lazy tiling algorithm, suitable for statistical experiments of CA rules. We have used it to perform experiments with Game of Life rules, and demonstrate that the Game of Life on the rhomb tiling is significantly different from that on the kite and dart tiling: it has longer lifetimes, lower ash densities, and higher soup growth.

Work is underway to investigate and classify the oscillators left in the ash.

References

[1] David Austin. (December 2005). Penrose tilings tied up in ribbons. *AMS Monthly Essays on Mathematical Topics*. `http://www.ams.org/featurecolumn/archive/ribbons.html`.

[2] D. Bakan. (1967). *On Method: Toward a Reconstruction of Psychological Investigation*. Josey-Bass.

[3] Elwyn R. Berlekamp, John Horton Conway, and Richard K. Guy. (1982). *Winning Ways for Your Mathematical Plays Volume 2: games in particular*. Academic Press.

[4] Jacob Cohen. (1988). *Statistical power analysis for the behavioral sciences*. Lawrence Erlbaum Associates, 2nd edition.

[5] N. G. de Bruijn. (1981). Algebraic theory of Penrose non-periodic tilings of the plane I and II. *Indagationes Mathematicae (Proceedings)*, 84:39–66.

[6] N. G. de Bruijn. (1986). Dualization of multigrids. *Journal de physique, Colloque C3*, 47:9–18.

[7] N. G. de Bruijn. (1996). Remarks on Penrose tilings. In R. L. Graham and J. Nesetril, editors, *The Mathematics of P. Erdös, volume 2*, pages 264–283. Springer.

[8] Martin Gardner. (October 1970). Mathematical games: The fantastic combinations of John Conway's new solitaire game "life". *Scientific American*, 223(4):120–123.

[9] Martin Gardner. (January 1977). Mathematical games: extraordinary non-periodic tiling that enriches the theory of tiles. *Scientific American*, 236(1):110–121.

[10] Branko Grünbaum and G. C. Shephard. (1987). *Tilings and Patterns*. W. H. Freeman.

[11] Margaret Hill, Susan Stepney, and Francis Wan. (2005). Penrose Life: ash and oscillators. In Mathieu S. Capcarrere, Alex A. Freitas, Peter J. Bentley, Colin G. Johnson, and Jonathan Timmis, editors, *Advances in Artificial Life: ECAL 2005, Canterbury, UK, September 2005*, volume 3630 of *LNAI*, pages 471–480. Springer.

[12] Marks R. Nester. (1996). An applied statistician's creed. *Applied Statistics*, 45(4):401–410.

[13] George Y. Onoda, Paul J. Steinhardt, David P. DiVincenzo, and Joshua E. S. Socolar. (1988). Growing perfect quasicrystals. *Physical Review Letters*, 60(25):2653–2656.

[14] Roger Penrose. (1978). Pentaplexity. *Eureka*, 39:16–32.

[15] P. Ramachandrarao, G. V. S. Sastry, L. Pandev, and Arvind Sinha. (1991). A novel algorithm for a quasiperiodic plane lattice with fivefold symmetry. *Acta Cryst.*, A47:206–210.

[16] Paul Rendell. (2002). Turing Universality of the Game of Life. In Andrew Adamatzky, editor, *Collision-Based Computing*. Springer.
[17] Marjorie Senechal. (1995). *Quasicrystals and Geometry*. Cambridge University Press.
[18] Sidney Siegel. (1956). *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill.
[19] Steven Silver. (February 2006). Life lexicon, release 25. `http://www.argentum.freeserve.co.uk/lex.htm`.
[20] George W. Snedecor and William G. Cochran. (1980). *Statistical Methods*. Iowa State University Press, 7th edition.
[21] Joshua E. S. Socolar and Paul J. Steinhardt. (1986). Quasicrystals. II. Unit-cell configurations. *Phys. Rev. B*, pages 617–647.