Size Matters: Layout Algorithms

September 11, 2024

1 Introduction

This document describes the layout algorithms implemented in Holes.py, which was used to run simulations reported in the article "Size Matters: Optimal Test Pit Size for Dispersed Archaeological Test Excavations" by Oakes and McLaren. See that article for background. We simulate archaeological digs to explore success of different test pit sizes, numbers of test pits, and layout strategies, with different sizes and types of archaeological sites. Our simulation takes place on a square field of dimension $n \times n$ ($100m \times 100m$ and $200m \times 200m$ in the simulations used in the article). The test pits sizes for simulations used in the article are $0.5m \times 0.5m$ (small pits) and $1m \times 1m$ (large pits). An archaeological site is placed randomly on the field, and pits dug according to a layout algorithm to see if they find the artefacts.

2 Hexagonal-Like Layout Algorithm

This is the algorithm used for the simulations reported in the article. We want a staggered grid pattern, as shown in figure 1. For a given total number of pits h we need to decide how many pits will be dug in our field horizontally (h_x), and how many vertically (h_y).

First some definitions:

- h is the total number of pits,
- h_x and h_y are the number of pits along the x axis and the y axis of the field respectively,
- n_x and n_y are the width and height of the field respectively (we have only tested square fields),
- a, b, and c are distances between pits as shown in figure 1,
- $pitSize$ is the length (and width) of the square test pits.

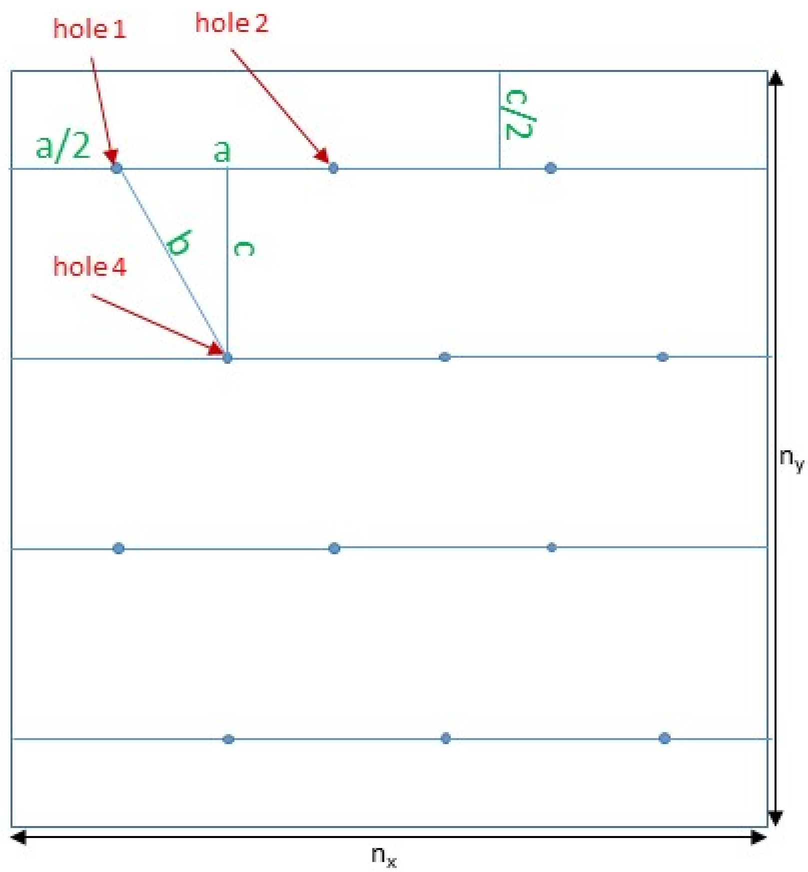

Figure 1: Layout algorithm for $h = 12$ test pits. A layout is hexagonal when $a = b$.

A layout is hexagonal when a and b shown in figure 1 are equal. A hexagonal layout has been shown to be optimal in certain circumstances. In the hexagonal-like algorithm we set a fixed horizontal border of $a/2$ and a fixed vertical border of $c/2$, and then choose values for h_x and h_y that make the layout almost hexagonal. In section 3 we vary the borders to create a truly hexagonal layout. The reason for fixing borders is to avoid anomalous results when a pit is close to the edge of the field, noting that the archaeological site is placed wholly within the field. In experiments the hexagonal-like and hexagonal algorithms had very similar success rates.

In this section, except where otherwise apparent, we treat pits as points, ignoring *pitSize*. We assume *pitSize* is much less than the size of the field.

We now derive some equations for h_x and h_y, starting with some equations evident from figure 1:

$$h = h_x \cdot h_y \tag{1}$$

$$n_y = c \cdot h_y \tag{2}$$

$$\begin{aligned} n_x &= a(h_x - 1) + a + \frac{a}{2} && (3) \\ &= a(h_x + \frac{1}{2}) && (4) \end{aligned}$$

$$a^2 = \left(\frac{a}{2}\right)^2 + c^2 \qquad \text{Pythagorean Theorem} \tag{5}$$

Now we write a formula for a in terms of n_x and h_x:

$$a = \frac{n_x}{h_x + \frac{1}{2}} \qquad \text{from 4} \tag{6}$$

Now we write a formula for a in terms of n_y and h_y:

$$\begin{aligned} a^2 &= \frac{a^2}{4} + c^2 \qquad \text{from 5} && (7) \\ \frac{3}{4}a^2 &= c^2 && (8) \\ a &= \frac{2c}{\sqrt{3}} && (9) \\ &= \frac{2\left(\frac{n_y}{h_y}\right)}{\sqrt{3}} && (10) \\ &= \frac{2n_y}{\sqrt{3}h_y} && (11) \end{aligned}$$

We equate the two formulae for a and find a quadratic equation for h_x:

$$\begin{aligned}
\frac{n_x}{h_x + \frac{1}{2}} &= \frac{2n_y}{\sqrt{3}h_y} && \text{from 6 and 11} && (12)\\
\sqrt{3}h_y n_x &= 2n_y \left(h_x + \frac{1}{2}\right) && && (13)\\
\sqrt{3}\frac{h}{h_x} n_x &= 2n_y \left(h_x + \frac{1}{2}\right) && \text{from 1} && (14)\\
\sqrt{3}h n_x &= 2h_x n_y \left(h_x + \frac{1}{2}\right) && && (15)\\
&= 2n_y h_x^2 + n_y h_x && && (16)\\
0 &= 2n_y h_x^2 + n_y h_x - \sqrt{3}h n_x && && (17)
\end{aligned}$$

We then solve for h_x using the quadratic formula:

$$\begin{aligned}
h_x &= \frac{-n_y + \sqrt{{n_y}^2 - 4 \cdot 2n_y \cdot (-\sqrt{3}h n_x)}}{2 \cdot 2n_y} && (18)\\
h_x &= \frac{-n_y + \sqrt{{n_y}^2 + 8\sqrt{3}n_x n_y h}}{4n_y} && (19)\\
\text{and} \qquad h_y &= \frac{h}{h_x} && (20)
\end{aligned}$$

We round the values to get integer values for h_x and h_y and recalculate a and c to reflect the rounded values:

$$\begin{aligned}
c' &= \frac{n_y}{round(h_y)} && (21)\\
a' &= \frac{n_x}{round(h_x) + \frac{1}{2}} && (22)\\
border_x &= \frac{a'}{2} && (23)\\
border_y &= \frac{c'}{2} && (24)
\end{aligned}$$

Then we layout the pits with the given borders, so that pits in a row are a' apart, the rows are c' apart, and every second row is indented (staggered) by $a'/2$.

If $border_x$ or $border_y$ is less than $pitSize/2$ the layout algorithm fails (note a pit location is specified by the coordinates of its centre).

3 Hexagonal Layout Algorithm

This section describes one way to implement a hexagonal layout algorithm. We do not claim that it is the best way. We start with the equations for a hexagonal-like layout with a horizontal border of $a/2$ and vertical border of $c/2$, calculating

h_x and h_y as in equations 19 and 20. Then we vary the borders so that it is exactly hexagonal. First fix $border_y$ and c to their original values and change $border_x$ and a:

$$\begin{aligned} c' &= \frac{n_y}{round(h_y)} && (25)\\ a' &= \frac{2c'}{\sqrt{3}} \qquad \text{from } 9 && (26)\\ border_x &= \frac{n_x - a'(round(h_x) - \frac{1}{2})}{2} && (27)\\ border_y &= \frac{c'}{2} && (28) \end{aligned}$$

It can be shown that for a square field, when h_x is similar to h_y, this increases the value of a' compared to the value in equation 22, shrinking $border_x$ (it appears to be true more generally). If $border_x < pitSize/2$, the pits won't fit (pits are specified by the coordinates of their centre). In that case we set $border_x$ it to its minimum value of $pitSize/2$ and vary the other values as follows:

$$\begin{aligned} a'' &= \frac{n_x - pitSize}{round(h_x) - \frac{1}{2}} && (29)\\ c'' &= \frac{\sqrt{3}a''}{2} \qquad \text{from } 9 && (30)\\ border_x &= \frac{pitSize}{2} && (31)\\ border_y &= \frac{n_y - c''(round(h_y) - 1)}{2} && (32) \end{aligned}$$

Our value for a may now be a' or a'', so we just refer to it as a. Likewise for c. If $a < pitSize$, $border_x < pitSize/2$, $c < pitSize$, or $border_y < pitSize/2$, we are unable to create a hexagonal layout using this method. Otherwise we layout the pits with the given borders, so that pits in a row are a apart, the rows are c apart, and every second row is indented (staggered) by $a/2$.

4 Staggering in the Vertical Direction

The hexagonal-like and hexagonal layout algorithms result in a staggered grid. This staggering is in the horizontal direction, with every second row indented. We can also stagger in the vertical direction, so that every second column is shifted down. This may improve on the hexagonal layout when the archaeological site becomes elongated rather than being circular. Adapting the hexagonal-like layout algorithm to do this requires a slight difference to the maths used to calculate h_x and h_y. The difference begins with a difference to equation 2, namely

$$n_y = c(h_y + \frac{1}{2}) \qquad (33)$$

Then similar maths to section 2 gives

$$
\begin{aligned}
4n_y h_x^2 + (2n_y - \sqrt{3}n_x)h_x - 2\sqrt{3}n_x h &= 0 && (34)\\
h_y &= \frac{h}{h_x} && (35)\\
c' &= \frac{n_y}{round(h_y) + \frac{1}{2}} && (36)\\
a' &= \frac{n_x}{round(h_x) + \frac{1}{2}} && (37)\\
border_x &= \frac{a'}{2} && (38)\\
border_y &= \frac{c'}{2} && (39)
\end{aligned}
$$

Equation 34 can be solved for h_x with the quadratic formula.

Then we layout the pits with the given borders, so that pits in a row are a' apart, the rows are c' apart, every second row is indented (staggered) by $a'/2$, and every second hole in each row is shifted down by $c'/2$.

The above is a staggered version of the hexagonal-like algorithm. It can also be used as a starting point for a staggered version of the hexagonal algorithm by following the steps outlined in section 3, using these equations for h_x and h_y. Replace equation 25 with equation 36 and replace equation 32 with:

$$
border_y = \frac{n_y - c''(round(h_y) - \frac{1}{2})}{2} \qquad (40)
$$

though note it will no longer really be hexagonal due to the staggering.

These algorithms were not used in the simulations for the article.

5 Random Layout Algorithm

The random layout algorithm chooses pit locations according to a pseudo-random number generator (of course computers cannot produce truly random results!). The pits do not overlap. This algorithm was not used in the simulations for the article.

6 Halton Layout Algorithm

The Halton layout algorithm lays out the pits according to a 2-dimensional scrambled Halton sequence, which appears random but does not cluster pits close to each other (which may happen in the random layout). This algorithm was not used in the simulations for the article.