

The Hopper Problem

Codehc

June 2023

0.1 Assumptions

All matrices and vectors are 1-indexed. \odot represents element-wise operations. Element-wise exponentiation of matrices represents raising the power of every element in a matrix to the corresponding element in the power matrix.

0.2 The Hopper Problem

NOTE THIS PARAGRAPH IS NOT CORRECT IGNORE THIS PART FOR NOW

You have n hoppers with a ball each. Every iteration, a random ball is selected. The hopper containing that ball doesn't change. Every other hopper doubles the amount of balls it has. On average, how long will it take for each hopper to have an equal amount of balls again?

This part is fine: A particle starts at the origin. At every distance from the origin, d , its chance of coming towards the origin is $\frac{2^d}{1+2^d}$ and its chance of moving away from the origin is $1 - \frac{2^d}{1+2^d}$. On average, how many iterations, where at each iteration it moves either towards or away from the origin, does it take to get back to the origin?

0.3 Path Representation

The most intuitive way to represent a path is with a series of arrows. Every iteration, you move either closer to the origin or further from the origin. For example, further further closer closer can be represented as $> > < <$. This is known as arrow representation.

A path can be represented as a matrix where the rows define the distance from the origin, the first column is moving away from the origin, the second column is moving towards the origin, and the values represent how many times the particle has moved towards or away from the origin at that distance. This is known as matrix representation. For example, the path $> > < <$ can be represented as such in matrix form:

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}$$

You can convert any path into a path matrix by first figuring out the distances from the origin at every point and mapping those to movements. For example, for the path $> > < > < > < <$:

Movements	>	>	<	>	<	>	<	<	
Distances	0	1	2	1	2	1	2	1	0

NOTE: The distances are calculated by adding the last movement to the last distance where $>$ is $+1$ and $<$ is -1 .

Then you can rewrite that as movement directions at distances:

Distances	Away	Closer
0	>	
1	>>>	<
2		<<<

Which can then be rewritten in the matrix form:

$$\begin{bmatrix} 1 & 0 \\ 3 & 1 \\ 0 & 3 \end{bmatrix}$$

The number of rows in our path is the maximum distance the particle reaches at any point along that path. Also known as m . We can take a submatrix of our path matrix in order to get a submatrix that represents how many times and at what distances the particle has moved *away from* or *closer to* the origin. This gives us the ability to apply a different matrix to our submatrices when using element wise matrix exponentiation. For example, the same path matrix can be split up into a away from the origin submatrix:

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}[:, 1] = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

The closer to the origin submatrix can be attained like this:

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}[:, 2] = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

Every path can be constructed by having 3 unique pieces of information for every movement: the iteration number at every movement, the distance from the origin at every movement, and the direction of every movement. However, only 2 of these 3 pieces of information need to actually be stored, the 3rd one can be deduced fairly easily. Arrow form holds the direction of every movement and the iteration number at every movement. The distance at every movement can be deduced by adding up all the movement directions up until the movement in question. Matrix form holds the distance from the origin at every movement and the movement direction of every movement. The rows represent the distances from the origin and the columns hold the movement directions. Deducing the iteration number at every movement is slightly more difficult so it won't be discussed here. In short, the path must be traced starting from the origin and ensuring that it never prematurely goes back to the origin before reaching every movement.

For our calculations regarding the probability of every path, the iteration number is wholly insignificant while the distance at every movement is significant. Matrix form is perfect since it holds the two necessary pieces of information.

0.4 Probability Vectors

Two probability vectors are needed. One for probabilities at different distances for moving away from the origin and one for moving towards the origin. The vector for moving towards the origin

can be defined formally like this:

$$P_{closer} = \left[\frac{2^{i-1}}{1+2^{i-1}} \right]_{m \times 1}$$

$1 \times m$ being the number of rows and columns in the vector. Once again, where m is the maximum distance from the origin. It follows that the vector for the probability of moving away from the origin can be defined like this:

$$P_{away} = \left[1 - \frac{2^{i-1}}{1+2^{i-1}} \right]_{m \times 1}$$

Expanded, these vectors look like this:

$$P_{closer} = \begin{bmatrix} \frac{1}{2} \\ \frac{2}{3} \\ \frac{3}{4} \\ \frac{4}{5} \\ \dots \end{bmatrix}$$

$$P_{away} = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{3} \\ \frac{1}{4} \\ \frac{1}{5} \\ \dots \end{bmatrix}$$

0.5 Probability Calculation

In order to find the probability of a given path, the path matrix, W , is split into two submatrices. One submatrix, C , holds the number of movements away from the origin and their distances from the origin and the other, A , holds the number of movements towards the origin and their distances from the origin. The probability matrix for moving towards the origin is then raised to the power of C using element-wise exponentiation, otherwise known as Hadamard exponentiation, denoted by \odot . The product of the elements in the generated matrix is then taken which is the probability for all the movements towards the origin. The same process is repeated with the probability matrix for moving away from the origin with the submatrix A . These probabilities are then multiplied together to get the total probability of the path. Formally written as:

$$P(W) = \prod_{i=1}^m \left(P_{closer}^{(\odot W[:,1])} \right) [j, 1] \times \prod_{i=1}^m \left(P_{away}^{(\odot W[:,2])} \right) [j, 1]$$

For example, let's calculate the probability of $> > < <$:

$$\begin{aligned}
P \left(\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \right) &= \prod_{i=1}^m \left(P_{closer} \left(\odot \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}_{[:,1]} \right) \right) [j, 1] \times \prod_{i=1}^m \left(P_{away} \left(\odot \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}_{[:,2]} \right) \right) [j, 1] \\
P \left(\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \right) &= \prod_{i=1}^m \left(\left[\frac{2^{i-1}}{1+2^{i-1}} \right]_{1 \times m} \left(\odot \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \right) \right) [j, 1] \times \prod_{i=1}^m \left(\left[1 - \frac{2^{i-1}}{1+2^{i-1}} \right]_{1 \times m} \left(\odot \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \right) \right) [j, 1] \\
P \left(\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \right) &= \prod_{i=1}^m \left(\left[\frac{1}{2} \frac{2}{3} \frac{3}{4} \frac{4}{5} \right] \left(\odot \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \right) \right) [j, 1] \times \prod_{i=1}^m \left(\left[\frac{1}{2} \frac{2}{3} \frac{3}{4} \frac{4}{5} \right] \left(\odot \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \right) \right) [j, 1] \\
P \left(\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \right) &= \prod_{i=1}^m \left[\frac{1}{2} \frac{2}{3} \frac{3}{4} \frac{4}{5} \right] [j, 1] \times \prod_{i=1}^m \left[\frac{1}{5} \frac{4}{3} \frac{3}{2} \frac{2}{1} \right] [j, 1] \\
P \left(\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \right) &= \left(\frac{1}{2} \times \frac{2}{3} \times 1 \right) \times \left(1 \times \frac{1}{3} \times \frac{1}{5} \right) \\
P \left(\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \right) &= \frac{1}{45}
\end{aligned}$$

0.6 Validating Paths

The validity of a path can be easily deduced by just examining its matrix representation. All valid paths have the closer column as identical to the away column just shifted 1 row down. Intuitively, this is because whatever set of moves the particle makes away from the origin must be replayed in reverse to get back to the origin. Where W is an arbitrary path matrix and m is the maximum distance from the origin, this can be represented as such:

$$W[1 : m - 1, 1] = W[2 : m, 2]$$

If the above statement is true for a matrix, then that matrix represents a valid path.