CHIPS ON A CHESSBOARD

SARAH QUINN, ANDREW L. TRATTNER, AND YUWEI ZHANG

Abstract. This paper analyzes a dynamic system of chips on a chessboard, characterized by certain initial settings and a redistribution rule. We study the redistribution of single stack initial configurations, giving bounds for the number of time steps needed to reach the steady state as well as the radius of occupied cells. We examine multiple stacks to prove that arbitrary finite initial configurations will reach steady state in finite time.

1. Introduction

In this paper, we study a dynamic system of chips on an infinite chessboard. Each grid cell initially contains some stack of chips, possibly zero. Chips are redistributed by an earthquake. The tremor repeatedly causes piles with four or more chips to move, one chip sliding to the four adjacent cells in compass directions as shown in Figure 1.

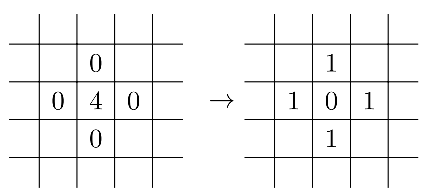

Figure 1. Example Redistribution of 4 Chips

We are interested in determining which initial chip configurations reach steady states and how long this redistribution process takes. Furthermore, what can be said about the configuration of chips in the steady state? Figure 2 is an example of a steady state reached after thousands of redistributions, starting with a single stack initial configuration of 3600 chips.

Date: November 7, 2016.

Sarah revised Section 1, 2, 3, 4, and 8. Andy revised Section 1, Section 2, Section 3, Section 4, Section 6 and Section 7. Yuwei revised the abstract, Section 4, Section 5, and Section 8.

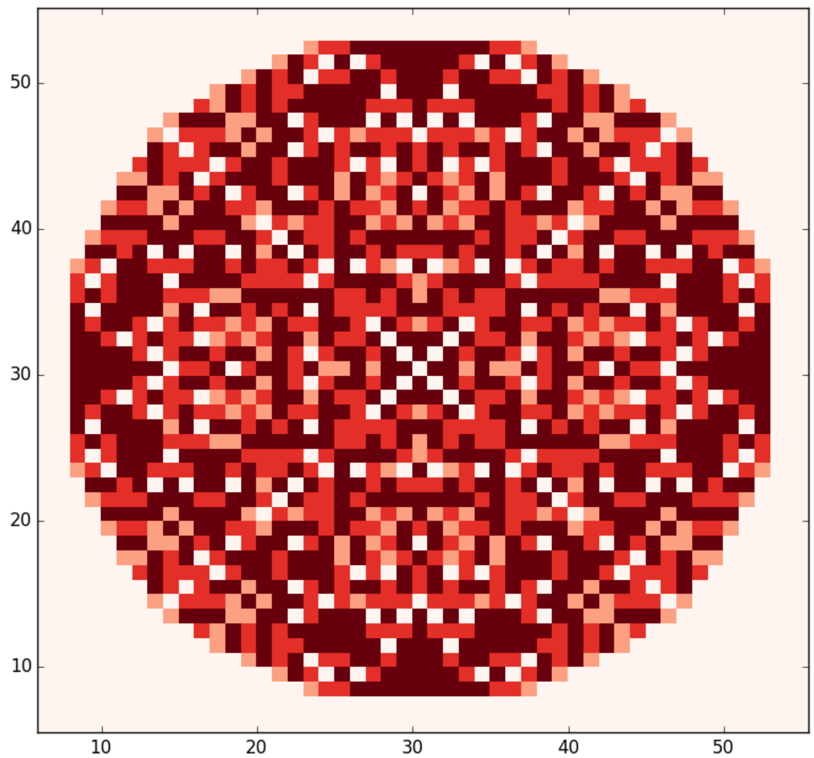

FIGURE 2. Example Steady State For Single Stack Of 3600 Chips. dark = 3, red = 2, pink = 1, white = 0 chips.

In this paper, we develop notation and terminology to represent this system in Section 2. We discuss time metrics in Section 3. In Section 4, we characterize the expansionary nature of the system and derive a layer method of analyzing the system. We develop a measure of steady state radius in Section 5. In Sections 6 and 7, we present step bounds for single and multi-stack initial chip configurations to conclude that all finite initial configurations reach steady state in finite time.

2. Notation

Let $N^t_{(a,b)}$ denote the number of chips in the cell at coordinate (a, b) at time t. For stacks with four or more chips at the current time step, the update rule moves one chip in each of the four cardinal directions, which we can represent as:

$$\begin{aligned} N^{t+1}_{(a,b)} &= N^t_{(a,b)} - 4 \\ N^{t+1}_{(a\pm1,b)} &= N^t_{(a\pm1,b)} + 1 \\ N^{t+1}_{(a,b\pm1)} &= N^t_{(a,b\pm1)} + 1. \end{aligned}$$

We can again refer to Figure 1 as a basic example of this redistribution process. We would call the left diagram the initial configuration and the right diagram in the figure the steady state.

Definition 2.1. When no cells contain 4 or more chips, we say that the chessboard has reached a **steady state**, or, equivalently, that the redistribution process has **terminated**.

To show the steps required to reach steady state in later analysis, we will need to use the concept of layers.

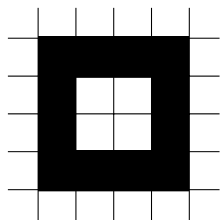

FIGURE 3. Example Of A Layer

Definition 2.2. A **layer** is a rectangular shell of adjacent cells that partitions the chessboard into an interior section and an exterior section. Cells composing the interior are called inner cells, located "inside" the layer, and cells composing the exterior are said to be "outside" the layer. An example is shown in Figure 3, where the dark filled "layer cells" constitute the layer. In this paper, all layers are assumed to be one cell thick.

3. TIME

There are two potential views of time: time steps as single cell redistributions, and time steps as whole board redistributions, where four chips are disbursed from every eligible cell simultaneously. Unless stated otherwise, we use the whole board redistribution view in this paper. But it is natural to question: is there a difference between the two views?

Clearly, the single cell redistribution view of time will require as many or more steps than the board redistribution view to terminate. We can easily see this using the concept of a temporary board.

Imagine that we are using two chessboards to chart redistribution, the original board and a temporary board where all cells contain zeroes. On each step, we look for all cells in the real board with four or more chips. For each of these, we subtract four from the corresponding temporary cell and add one to the temporary adjacent cells per the redistribution rule. To complete an update, superimpose the temporary board on the real board.

Essentially, this procedure captures single redistributions and maps them to whole board redistribution. So if a steady state can be computed for one view of time, that same steady state will be computed by the other view.

There is an interesting case in which the two views differ, but it involves infinite chips and cannot be simulated fully by a computer.

Consider a configuration where we have a 4 in the first square, followed by a ray of 3's in one of the 4 compass directions, $4333\cdots$. When we have such a setting, the process would not terminate for either view because redistributions would cascade right indefinitely.

4	3	3	3	···

FIGURE 4. A Non-Terminating Initial Configuration

Now the interesting case arises when we modify this non-terminating configuration to have an alternating $434343\cdots$ pattern.

4	3	4	3	···

FIGURE 5. A Special Configuration

With the board redistribution view, all the fours simultaneously redistribute, then all the threes, and we terminate in two time steps. But clearly, the single cell redistributions would take infinitely many steps since there are infinitely many cells which require redistributing.

4. Always Expanding Principle and Layer Method

In this section, we develop tools which will help us prove bounds on termination for single stack initial configurations and multi-stack configurations.

Consider how a layer redistributes chips. As shown in Figure 6, chips shuffle out, into the center, or among the layer cells themselves (not pictured).

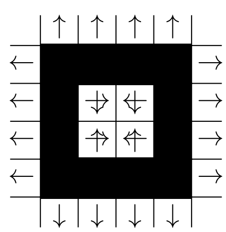

FIGURE 6. Outward Redistribution Of A Layer

We call the following theorem the "Always-Expanding Principle" (AEP) and we easily prove it by observing the behavior of the layer.

Theorem 4.1. *The number of chips moving outward from a layer is greater than or equal to the number of chips moving inward during any given time step.*

Proof. By the redistribution rule, cells in the layer will redistribute chips in the four cardinal directions. This means all layer cells redistribute two chips to neighboring layer cells, and two chips either outward or inward. All the layer cells except the corners must redistribute equal amounts outward and inward. The corner cells only redistribute outward and within the layer itself. □

AEP is simple but useful. We call the following corollary the "Layer Method" and give proof because it is not directly obvious.

Corollary 4.1.1. If N chips are initially located inside of a layer of empty cells, and all outside cells are also empty, then at most $3N$ chips will ever visit the layer cells over time.

Proof. The only chips in the system reside inside the layer and there are N of these. Then after some redistributions, at most N chips might reside within layer cells. Consider how these chips might move inward or outward from the layer over time, to the adjacent layers.

At most all of the N chips are eventually pushed from the layer. We know that at most half go inward to an interior layer by AEP. At most, these chips might all come back to the layer after some redistributions. This process could repeat over time and eventually we would have at most N chips re-visiting the layer from the inside, since

$$\frac{N}{2} + \frac{N}{4} + \frac{N}{8} + \ldots = N\left(\sum_{k=1}^{\infty} \frac{1}{2^k}\right) = N,$$

and we know that finite chip counts cannot be subdivided indefinitely.

AEP provides no upper bound on chips moving outward, so we are limited only by N. Assume all N chips eventually get pushed out of the layer. Then at most half of these can return to the layer by AEP. The same analysis as above applies, and N chips total might revisit the layer over time from the outside.

So we have N chips coming into the layer initially, then N return visits from the interior and N return visits from the exterior. In total, this yields $3N$ chips visiting the layer over all time. □

5. Radius

From Figure 2 in Section 1, we see that the steady state configuration is somewhat circular and symmetric with respect to the axes. It is natural, then, to consider the concept of the radius of the steady state, which will help us in our goal of bounding the number of steps to termination for a single stack.

Definition 5.1. The **radius** r of the steady state configuration refers to the distance from the furthest cell on the axes that is ever reached before termination to the origin where the initial single pile is placed.

Let each unit cell be of length 1. If the cell is at coordinate $(a, 0)$, then $r = a$. For example, Figure 1 in Section 1 shows the steady state of $r = 1$ resulting from an initial pile of 4 chips.

We are interested in finding the bounds for the radius as a step towards finding the bounds for the number of steps needed for termination. First, we prove a lemma which would be useful for obtaining the bounds on the radius.

Lemma 5.2. *The steady state configuration does not contain adjacent zeros.*

Proof. Suppose for the sake of contradiction that we have both 0's on two adjacent cells reached within the radius. As the cells had chips on them at some step before termination, both cells having 0 chips can only be a result of redistribution. It must be that they redistributed and terminated to become zero either at the same step or at different steps.

(1) If they terminated at the same step, they would each have 1 chip afterwards, a contradiction.
(2) If they terminated at different steps, the pile that terminated last would have moved 1 chip to the other pile that terminated first, leaving a 1 there at the end, a contradiction.

Thus, the steady state does not have adjacent zeros within its configuration, but only singular 0's. □

Now, for the simplicity of notations, let N denote the number of chips in the initial pile, $N^0_{(0,0)}$.

Lemma 5.3. *The radius of the steady state configuration has an upper bound of* $\sqrt{N}$.

Proof. As the chips move to the four compass directions at each redistribution, the steady state configuration at least covers a diamond, which is in fact a square with a diagonal of length $2r$ by the symmetry of the redistribution process. Below shows a diamond configuration for $r = 2$, and an example of the steady state from an initial pile of 16 chips:

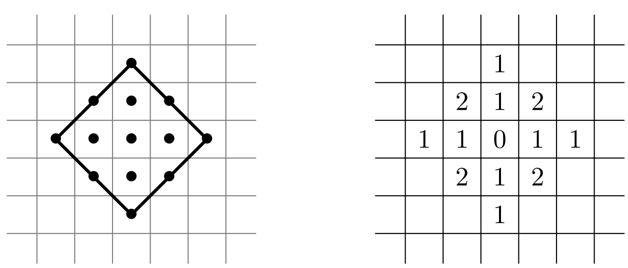

FIGURE 7. Steady State Configuration Covers At Least A Diamond

To see this, we use induction from the initial pile at the origin, a_1. We first examine the upper right quadrant. By a redistribution from a_1, cells b_1 and b_2 are reached at the same step. Subsequently, c_1, c_2, and c_3 are reached at the same step by a redistribution from b_1 and b_2. The same argument applies for the other three quadrants. By induction, we see that the cells on the four sides of the diamond configuration will be reached at the same step.

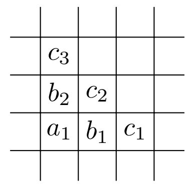

FIGURE 8. Illustration in the Upper Right Quadrant

Hence, since r is the furthest distance from a cell on the axes that is ever reached in the process, the total number of squares reached at the steady state is at least the number of squares in the diamond shape with diagonals of length $2r$, which is $r^2+(r+1)^2$. For $r = 2$, the number of squares reached is $r^2+(r+1)^2 = 2^2+3^2 = 13$.

In addition, from Lemma 5.2, we see that we would no adjacent 0's but only at most single 0's in alternate squares in the steady state configuration. This means that at least half of the squares have at least one chip on it. Thus, we have the lower bound for N:

$$\begin{aligned} N &\geq \frac{1}{2}(r^2 + (r+1)^2), \\ r^2 + (r+1)^2 &\leq 2N, \\ r^2 &< N, \\ r &< \sqrt{N}. \end{aligned}$$

□

Lemma 5.4. *The radius of the steady state configuration has an lower bound of* $\sqrt{\dfrac{N}{12}} - \dfrac{1}{2}$.

Proof. The steady state configuration is bounded by a square with side length of $2r$. An example of the bounding square on steady state configuration from an initial pile of 28 chips is illustrated below.

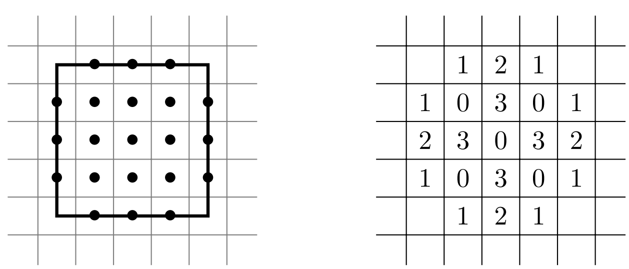

FIGURE 9. Steady State Configuration as Bounded by A Square

To see this, consider a pile of chips in a square, z_1, in the upper right quadrant. To reach z_1 at coordinate (a, b), it must have come from the redistribution of either the adjacent square below, y_1 at $(a, b-1)$, or the one to the left of it, y_2 at $(a-1, b)$, both of which have at least one coordinate that is as large as that of z_1.

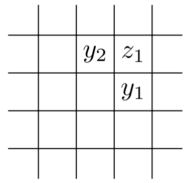

FIGURE 10. Illustration in the Upper Right Quadrant

By the definition of radius and induction on the edge squares in all four quadrants using the same reasoning, we see that the steady state configuration is bounded by a square with side length of $2r$, which covers $(2r+1)^2$ squares.

We can have at most 3 chips per pile in the steady state configuration. Thus,

$$N \leq 3(2r+1)^2,$$
$$r + \frac{1}{2} \geq \sqrt{\frac{N}{12}},$$
$$r \geq \sqrt{\frac{N}{12}} - \frac{1}{2}.$$

□

Hence, we have the proved following theorem:

Theorem 5.5. *The radius r of the steady state configuration for a single initial pile of chips is bounded by*

$$\sqrt{\frac{N}{12}} - \frac{1}{2} \leq r < \sqrt{N}, \qquad \text{where } r \text{ is an integer.}$$

6. Single Stack Step Bound

To bound the time steps required for a single stack initial configuration to terminate, we first make an observation about single stack configurations in general.

Lemma 6.1. *Given a single stack initial configuration $N^0_{(a,b)}$, all cells on the board except the central cell (a,b) contain less than eight chips at any given time step.*

Proof. All cells except (a,b) start with zero chips. They never gain more than four chips at once by the redistribution rule. So if at any time they have four or more chips, these are immediately redistributed on the next time step to bring the total chip count below four. It is possible that up to four chips are simultaneously added, but the total then remains below eight. □

We will use this lemma to prove the following result.

Theorem 6.2. *For a single stack of N chips, at most $\frac{3}{4}N + 4r$ steps are required to reach a steady state configuration of the board, where r is the steady state radius.*

Upper bounding $r = \sqrt{N}$, we show the theoretical bound versus actual steps for up to 3600 chips in Figure 11.

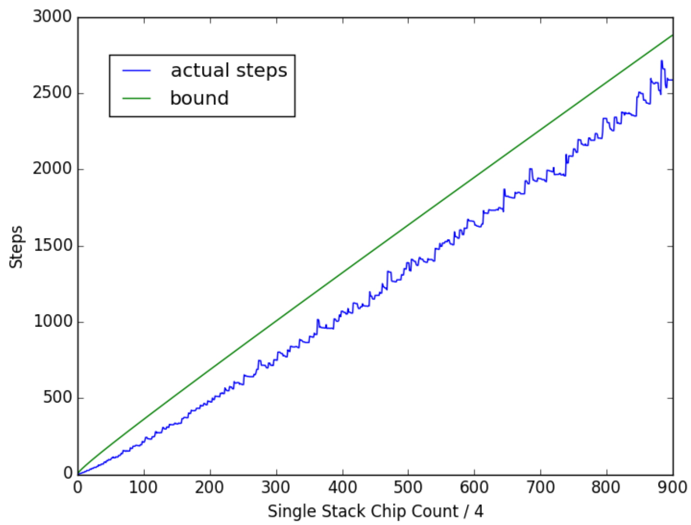

FIGURE 11. Steps To Termination

Proof. To simplify our analysis, we consider $N^0_{(a,b)} = 0 \pmod 4$ without loss of generality.

First, we prove the $\frac{3}{4}N$ component. Consider the layer immediately outside of the central cell. By the Layer Method, we have at most $3N$ chips visiting this layer. These chips initially came from the central cell. Then by the redistribution rule, $\frac{3}{4}N$ redistribution steps are required in total for the central cell.

All that remains is the $4r$ component. Once all of the redistributions are finished in the central cell, the surrounding cells within the steady state radius might continue redistributing. If we consider these surrounding cells in isolation from each other, each can redistribute at most once since each has less than eight chips by Lemma 6.1. So without isolation, this means that two redistributions might occur for each cell, once by virtue of its own chips and once by virtue of surrounding cell contributions.

In the worst case, a cascade of redistributions could begin on the outer periphery of occupied cells, propagate inward radially, and back outward once more.

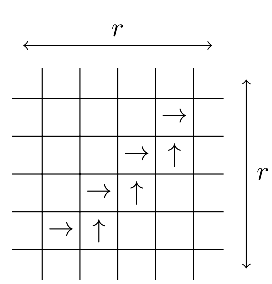

FIGURE 12. Example Radial Redistribution

Radial path lengths are at most $2r$ by the geometry of redistributions as illustrated in Figure 12. □

Corollary 6.2.1. By direct implication, we have finite termination time for all single stack initial settings of $N < \infty$ chips.

7. Arbitrary Finite Piles

Theorem 7.1. *Any finite setting* $\{N^0_{(i,j)} \mid N^0_{(i,j)} \geq 0 \ \forall \ |i|, |j| < \infty \text{ else } 0\}$ *is guaranteed to terminate in finite steps.*

Proof. Take the maximal and minimal i and j such that the rectangle formed by those bounding coordinates contains all nonzero stacks at time $t = 0$. Let N be the number of chips in the tallest stack.

Consider a separate empty board where we take the bounding rectangle and fill every cell with N chips. We show that this worst-case scenario is guaranteed to terminate, hence our original setting must also terminate.

Consider the periphery cells of the bounding rectangle. Call the larger side length s. Then there are s^2N chips inside the layer. Applying the Layer Method, at most $3s^2N$ chip visits to the layer occur over all time. Similarly, at most $3s^2N$ visits occur with respect to any other layer within the boundary, hence also outside of the boundary.

So we simply execute more than $S = \frac{3}{4}s^2N$ steps for every cell which ever contains chips, which we know is bounded by some radius r from the bounding rectangle's edges, through application of Lemma 5.2 analysis similar to the proof of Lemma 5.3.

In total, there will be at most $C = (s + 2r)^2$ cells which ever contain chips. So the total number of steps until termination will be less than S times C. □

8. Conclusion

In our study of a dynamic system of chips on a chessboard, we have found that with initial settings of a finite number of chips, the system always reaches a steady state and terminates. For an initial single stack of chips, we characterized the system by giving the bounds for the radius of the steady state configuration, as well as the upper bound for the number of steps it takes to terminate.

The resulting chips configurations show symmetry along several axes as seen from our simulation plots, and display interesting fractal-like patterns which are abound in nature. The study might also be of interest for applications in earthquake modeling, network flow problems, biological processes, and many others.

Further research could explore variations of the update rule, such as considering redistribution rules that move chips in each of the 8 directions instead of 4, and ones that move chips probabilistically instead of deterministically, with each chip moving to each direction with a different probability.