Article

Not peer-reviewed version

Modified Engel Algorithm and Applications in Absorbing/Non-Absorbing Markov Chains and Monopoly Game

Jeff Chak Fu Wong * and Chun He Liu

Posted Date: 4 July 2025

doi: 10.20944/preprints202507.0451.v1

Keywords: engel algorithm; markov chains; monopoly; classification

doi:10.20944/preprints202507.0451.v1

Article

Modified Engel Algorithm and Applications in Absorbing/Non-Absorbing Markov Chains and Monopoly Game

Chunhe Liu and Jeff Chak Fu Wong *

Department of Mathematics, The Chinese University of Hong Kong, Hong Kong SAR; 1155160246@link.cuhk.edu.hk
* Correspondence: jwong@math.cuhk.edu.hk

Abstract

The Engel algorithm was created to solve chip-firing games and can be used to find the stationary distribution for absorbing Markov chains. Kaushal et. al. developed a MATLAB-based version of the generalized Engel algorithm based on Engel's probabilistic abacus theory. This paper introduces a modified version of the generalized Engel algorithm, which we call the modified Engel algorithm or the mEngel algorithm, for short. This modified version is designed to address issues related to non-absorbing Markov chains. It achieves this by breaking down the transition matrix into two distinct matrices, where each entry in the transition matrix is calculated from the ratio of the numerator and denominator matrices. In a nested iteration setting, these matrices play a crucial role in converting non-absorbing Markov chains into absorbing ones and then back again, thereby providing an approximation to the solutions of non-absorbing Markov chains until the distribution of a Markov chain converges to a stationary distribution. Our results show that the numerical outcomes of the mEngel algorithm align with those obtained from the power method and the canonical decomposition of absorbing Markov chains. We provide an example, such as Torrence's problem, to illustrate the application of absorbing probabilities. Furthermore, our proposed algorithm analyzes the Monopoly transition matrix as a form of non-absorbing probabilities based on the rules of the Monopoly game, a complete information dynamic game, particularly the probability of landing on the Jail square, which is determined by the order of the product of the movement, jail, chance, and community chest matrices. There are more than two players in one game, and the last player who is not bankrupt wins. Strategies in the game include whether to spend as long as possible in prison to avoid paying tolls to other players, known as long jail, or to leave prison immediately to buy as much land as possible, known as short jail. The long jail strategy, the short jail strategy, and the strategy of getting out of Jail by rolling consecutive doubles three times have been formulated and tested. In addition, choosing which color group to buy is also an important strategy. By comparing the probability distribution of each strategy and their profit return on each property and the colour group property, we find which one should be used when playing Monopoly. In conclusion, the mEngel algorithm, implemented using R codes, offers an alternative approach to solving the Monopoly game and demonstrates practical value.

Keywords: engel algorithm; markov chains; monopoly; classification

---

1. Introduction

The chip-firing game (CFG), also known as a probabilistic abacus, is a dynamic process described in [8,9]. It involves a graph with a set of finite vertices and edges, on which multiple tokens, or "chips", are placed. In this game, if the number of chips on a vertex is greater than or equal to the degree of that vertex, it can be "fired" by moving one chip along each incident edge and placing it on the adjacent vertex. When a vertex is fired, its chip count is reduced by its degree, while its neighbors' chip counts are increased by one. The CFG is a directed weighted multigraph with a set of vertices and multiple

 doi:10.20944/preprints202507.0451.v1

edges sharing the same vertex. Firing a vertex means sending one chip along each outgoing edge from that vertex. It can be formulated as the transition digraph for an absorbing Markov chain with transition probabilities. Engel provided another method for solving chip-firing games based on the canonical decomposition of absorbing Markov chains in [21]. To start the game, a pile of pre-assigned chips is given. An indicated number of chips is set at each transient state, and zero chips are set at each absorbing state. Then, a vertex k is considered a critical position, and the number of chips g_k is placed on each adjacent vertex neighborhood of that vertex k based on the transition probabilities between adjacent vertices and the vertex k. This is referred to as "critical loading" in Kaushal et al.'s work [13,14]. If the number of chips at the critical position (vertex k) is not enough to fire, an additional chip (or additional chips) will taken from the pile and placed on the vertex k, and then fired to each adjacent vertex. This process continues as long as the critical position does not reappear in the transient states in the transition diagram. Once the number of chips in all the transient states match the initial number of chips in those states, then the game stops [15]. Chips placed in absorbing states can have a value of either nonzero or zero. By using a normalization procedure, we can determine the stationary distribution of the Markov chains from the transient states to the absorbing states based on the number of chips in each absorbing state. Engel's method involves representing each absorbing Markov chain as a directed graph in order to find a solution. This method is applicable to any Markov chains with at least one absorbing state.

The generalized Engel algorithm, as proposed by Kaushal et al. [13,14], is designed to solve the CFG by focusing on the transition states and absorbing states of a Markov chain. This algorithm is known for its higher accuracy compared to other methods but does depend on several factors. Kaushal et al. [13] implemented the Engel algorithm into a computer program to expand its application to more complex absorbing Markov chains, which they termed the generalized Engel algorithm. They achieved this by splitting the transition matrix into two separate matrices: the numerator matrix containing the numerators of entries in the transition matrix, and the denominator matrix containing the denominators of entries in the transition matrix as weights or for scaling. The numerator matrix represents the proportional distribution values of each transition state, while the denominator matrix represents the capacity of each transition state. They used the numerator matrix to simulate "firing" and the denominator matrix to reduce the divided chips. Our modified algorithm has a broader application and can provide process values. For example, in a directed graph, we can construct the set of configurations obtained from the CFG by a sequence of firings. Through process values, we can track the movement of every chip, determine how many chips pass through states, and record each configuration with the updated chips.

Our paper introduces a modified version of the generalized Engel algorithm, which we call the mEngel algorithm. This modified algorithm is designed to solve both absorbing and non-absorbing Markov chain problems. The key difference between the generalized Engel algorithm and our modified version is how it handles non-absorbing Markov chains. We accomplish this by reformulating the transition matrix decomposition into two separate matrices, similar to the approach in [13,14]. However, we further split the numerator matrix into nine sub-block matrices, representing a transformation from non-absorbing Markov chains to absorbing Markov chains and back to non-absorbing Markov chains. The mEngel algorithm works by determining the number of chips at each vertex and then using the algorithm to find all probability distribution integral values and store them in each corresponding vertex. The process continues until it stops when the tolerance is satisfied. Essentially, the mEngel algorithm provides approximate solutions for non-absorbing Markov chains. The study of Monopoly as a Markov process and its solution solver using the power method and the canonical decomposition of absorbing Markov chains has been investigated by several researchers such as [1–3,10,11,18–20,23,26], among others. Various strategies related to Monopoly's Jail were explored by analyzing the size of the transition matrix of non-absorbing Markov chains [4,5,7,17], as well as the long and short jail scenarios [24]. The mEngel algorithm was utilized to compare the ranking results of the long and short jail strategies.

 doi:10.20944/preprints202507.0451.v1

The rest of the paper is organized as follows. Section 2 introduces the transition diagram for the non-absorbing Markov chain model using a nested iterative setting. Section 3 first introduces the formulation of the transition probability matrix for absorbing/non-absorbing Markov chains and then provides a proof of the mEngel algorithm for solving the non-absorbing Markov chain problem using a nested iterative approach. This method is associated with the direct approaches of the power method and the canonical decomposition of absorbing Markov chains. The flow of the mEngel algorithm is provided, along with detailed descriptions of Algorithms 1 - 3. Section 4 presents the numerical results of the mEngel algorithm in solving for the absorbing probabilities in Torrence's problem [22,25]. Finding the stationary probabilities of the non-absorbing Markov chains based on the rules of the Monopoly game regarding Jail are presented, i.e., the probability of landing on each state is found. Emphasis is placed upon the Monopoly player regarding "Go to Jail" for three reasons, 1. Landing on "Go to Jail", 2. Drawing a "Go to Jail" card, 3. Rolling doubles three times, using the long jail and short jail strategies [4,24], which can be formulated as the 43×43 model and the 41×41 model, respectively, where the number of the player's turns to profit is studied and ranking based on the properties of the same colour group are examined. Another model, known as the 123×123 model [5,17], is also presented in this paper. It involves getting out of Jail by rolling consecutive doubles three times. Section 5 draws conclusions and identifies opportunities for further research.

2. From the Complete Graph to the Modified Graph

The CFG, a discrete dynamic model, can be played on undirected and directed graphs. In this paper, we focus on directed graphs and their relationship with Markov chains, which can be viewed as an absorbing Markov chain. A directed graph $G(V,E)$ is defined with vertices $V(G)$ and edges $E(G)$. Each vertex $v \in V(G)$ has a capacity d. An arbitrary number of chips g_v are placed on each vertex. If the number of chips g_v exceeds the capacity d, i.e., $g_v \geq d$, some chips will be passed to each neighboring vertex along the outgoing edges, called firing [13–15].

Suppose that a CFG starts with critical loading and ends with critical loading reoccurring. In that case, the chip distribution of this CFG is equal to the stationary distribution of the absorbing Markov chain. The critical loading occurs when each vertex v_i has one less chip than it needs in order to fire. In other words, $g_{v_i} = d_i - 1$. The critical loading of each CFG will reoccur, which has been proven by [15]. Hence, the Engel algorithm is complete, which means it always provides the solution for the absorbing Markov chain.

Let us have the directed graph $G(V,E)$ of a Markov Chain, represented as a transition diagram as shown in Figure 1. Assume that every jump passes through this directed graph. We take the vertex and edge of it to create a new directed graph, as shown in Figure 2.

Let the new graph be $\widehat{G}$. We have the vertex set $V(\widehat{G})$ and the edge set $E(\widehat{G})$, where $V(\widehat{G}) = \{s\} \cup V(G) \cup V(G^{'})$, and $V(G^{'})$ is a set with the same size as $V(G)$, for $\widehat{e}_{ij} \in E(\widehat{G})$, $e_{ij} \in E(G)$, $\widehat{e}_{ij} = e_{ij}$ if $v_i \in V(G), v_j \in V(G^{'})$, $\widehat{e}_{ij} = 0$ if $v_i, v_j \in V(G)$ or $v_i, v_j \in V(G^{'})$.

The mEngel algorithm proposed for solving the absorbing Markov chain as well as the non-absorbing Markov chain consists of three steps:

Step 1 We begin with the state space of a non-absorbing Markov chain denoted by S, then create another state space of an absorbing Markov chain with the same number of states as in the original chain of the transition diagram, denoted by S_1. In this new absorbing chain, the state space S_1 is not fully connected. Then, we introduce the additional state space denoted by S_2, which is also not fully connected. All states in S_2 are recurrent, meaning they eventually return to themselves. In general, a state is considered recurrent if, whenever we leave that state, we will return to it in the future with probability one. S_2 cannot go back to S_1.

Step 2 We introduce an artificial state s in S as a starting state. This state is not part of the original Markov chain but is added to the algorithm. The transition probabilities from S to each state in S_1 are set to probability distribution values (i.e., certain transitions). In contrast, the transition probabilities from S to each state in S_2 are all set to zeros (i.e., impossible transitions).

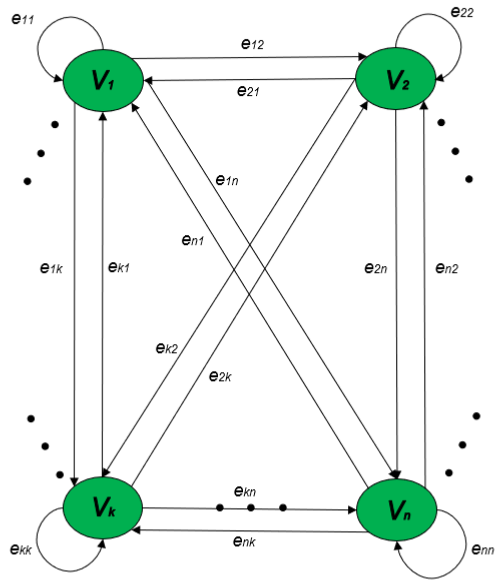

Figure 1. The directed graph of the original Markov chain

Step 3 The stationary distribution of this modified Markov chain (including S, S_1, and S_2) is equivalent to the original non-absorbing Markov chain. The proof of this equivalence is provided in Section 3.

In Figure 2, we pick a one-time jump (the player moves one time (one turn of the game)) of the Markov chain. With this time limit (a chain with only one jump), the Markov chain becomes absorbing. Then, we can iterate this absorbing Markov chain to get the stationary distribution of the whole Markov chain. The mEngel algorithm allows us to get the probability of the evolution process through an iterative process, i.e.,

$$\{S \to S_1 \to \ S_2\} \to \{S \to S_1 \to S_2\} \to \cdots .$$

For example, one observes how often chips pass through one state before the game ends by a sequence of configurations.

3. Structure of the mEngel Algorithm

The transition probability matrix P of the Markov chain for the mEngel algorithm can then be represented in the following canonical form, assuming time homogeneity, that is

$$P = \left[\frac{n_{ij}}{d_{ij}}\right],\ i,j = 1,\cdots,n$$

where $N = [n_{ij}]$ is a state (nominator) matrix, $D = [d_{ij}]$ is a scaling (denominator) matrix, and n is the total number of states. Note that $P = N/D$ is a non-absorbing matrix and $\widehat{P} = \widehat{N}/\widehat{D}$ is an absorbing matrix. The distinction between P and $\widehat{P}$ arises from the operation of Algorithms 1 - 2. Specifically, when we input P into Algorithm 2, it generates N and D as outputs. Similarly, when we input $\widehat{P}$, the algorithm produces $\widehat{N}$ and $\widehat{D}$. It is important to note that in our computations, we only utilize $\widehat{P}$.

The mEngel algorithm's transition diagram is composed of three state processes: 1) $S = \{s\}$ is a source state; 2) $S_1 = \{1,\cdots,n\}$ is a set of original states from the given problem; 3) $S_2 = \{1',\cdots,n'\}$ is a set of fictitious states for an mEngel algorithm setting. We use labels (e.g., s) instead of full names for vertices V_s, i.e., $S = \{s\} = \{V_s\}$. The transition diagram is shown in Figure 2.

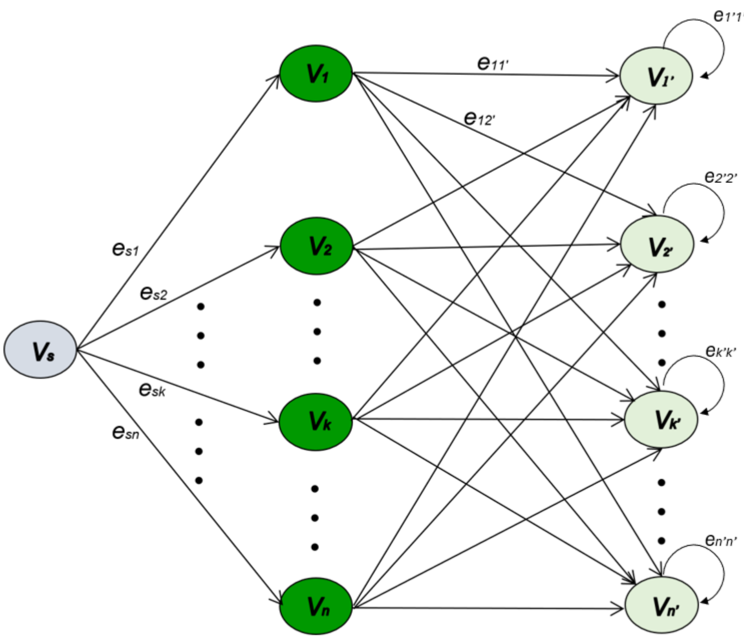

Figure 2. The directed graph of the reconstructed Markov chain

Define

$$\widehat{N} = \begin{array}{c} \\ s \\ \{1,\cdots,n\} \\ \{1',\cdots,n'\} \end{array} \begin{array}{c} \begin{array}{ccc} s & \{1,\cdots,n\} & \{1',\cdots,n'\} \end{array} \\ \begin{bmatrix} \mathbf{0}_{1\times 1} & \boldsymbol{d}^{(k)}_{1\times n} & \mathbf{0}_{1\times n} \\ \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times n} & \mathrm{ceil}(c\cdot P) \\ \mathbf{0}_{n\times 1} & \mathbf{0}_{n\times n} & I_n \end{bmatrix} \end{array} \tag{1}$$

and

$$\widehat{D} = \begin{array}{c} \\ s \\ \{1,\cdots,n\} \\ \{1',\cdots,n'\} \end{array} \begin{array}{c} \begin{array}{ccc} s & \{1,\cdots,n\} & \{1',\cdots,n'\} \end{array} \\ \begin{bmatrix} m^{(k)}_{1\times 1} & \boldsymbol{m}^{(k)}_{1\times n} & \boldsymbol{m}^{(k)}_{1\times n} \\ c\cdot\mathbf{1}_{n\times 1} & c\cdot\mathbf{1}_{n\times n} & c\cdot\mathbf{1}_{n\times n} \\ \mathbf{1}_{n\times 1} & \mathbf{1}_{n\times n} & \mathbf{1}_{n\times n} \end{bmatrix} \end{array} = \begin{pmatrix} \boldsymbol{m}^{(k)}_{1\times 2n+1} \\ c\cdot\mathbf{1}_{n\times 2n+1} \\ \mathbf{1}_{n\times 2n+1} \end{pmatrix} \tag{2}$$

where $m^{(k)}_{1\times 1} = \sum_{i=1}^{n} d^{(k)}_i$, $\boldsymbol{m} = \begin{bmatrix} \sum_{i=1}^{n} d^{(k)}_i & \sum_{i=1}^{n} d^{(k)}_i & \cdots & \sum_{i=1}^{n} d^{(k)}_i \end{bmatrix}_{1\times n}$ and c is a suitably sized scaling number.

1. The ceil operation is used here to ensure that every element of the matrix $\widehat{N}$ is an integer since the distributing chips are in an integer process. For example, if an element of the transition matrix P for a non-absorbing Markov chain/absorbing chain is 0.949 and $c = 100$, then the unrounded value will be 94.9. Thus, the rounding-up value is 95. One can choose c as 1000, and the rounding-up procedure is avoided. Similarly, each corresponding entry of the ratio of $\widehat{N}$ and $\widehat{D}$ is equal to each position of the transition matrix P. Hence, $\widehat{D}$ is a matrix scaled by c.
2. From S to $\{1,\cdots,n\}$, each arrow means a directed movement, where the number of chips being distributed is reshuffled and added. At the end of the iteration process, the number of configurations (chips) between the initial and final transient states remains the same.
3. From $\{1,\cdots,n\}$ to $\{1',\cdots,n'\}$, the original transition matrix P is scaled by c. As a result of this rounding-up process, the chips in a vertex are distributed to its neighboring vertices based on the probabilities obtained from rounding up the values of $c\cdot P$ to the nearest integers.
4. From $\{1',\cdots,n'\}$ to $\{1',\cdots,n'\}$, a set of transient states will be forced into a set of recurrent states, e.g., a state j is called an absorbing state if, with probability 1, the process will eventually

return to j after it leaves j. Hence, this is crucial for transforming the transient state into the absorbing state. Typically, a self-loop at each state means that a state of a Markov chain is called absorbing if, once entered, it cannot be left. Chips is stored at each vertex.

5. At the end of the iterative process, within the pre-assigned tolerance and the number of iterations, the stationary distribution for non-absorbing Markov chains is obtained by summing all chips from all vertices, i.e., $m_{1\times 1}^{(k)} = \sum_{i=1}^{n} d_i^{(k)}$, using the normalization procedure, i.e., $d_i^{(k)} / \sum_{i=1}^{n} d_i^{(k)}$ for all i.

Given the initial probability distribution $\boldsymbol{d}^{(0)}$. Our aim here is to show that

$$\text{mEngel}(\boldsymbol{d}^{(0)}) = \boldsymbol{d}^{(1)} = \boldsymbol{d}^{(0)}P \to \text{mEngel}(\boldsymbol{d}^{(1)}) = \boldsymbol{d}^{(2)} = \boldsymbol{d}^{(1)}P \to$$

$$\text{mEngel}(\boldsymbol{d}^{(2)}) = \boldsymbol{d}^{(3)} = \boldsymbol{d}^{(2)}P \to \cdots \to \text{mEngel}(\boldsymbol{d}^{(k)}) = \boldsymbol{d}^{(k+1)} = \boldsymbol{d}^{(k)}P.$$

For each iteration in the mEngel algorithm, we will get the same probability distribution for a non-absorbing/absorbing Markov chain using the power method, i.e.,

$$\boldsymbol{d}^{(1)} = \boldsymbol{d}^{(0)}P \to \boldsymbol{d}^{(2)} = \boldsymbol{d}^{(1)}P \to \boldsymbol{d}^{(3)} = \boldsymbol{d}^{(2)}P \to \cdots \to \boldsymbol{d}^{(k+1)} = \boldsymbol{d}^{(k)}P.$$

What needs to shown is that

$$\text{mEngel}(\boldsymbol{d}^{(1)}) = \boldsymbol{d}^{(2)} = \boldsymbol{d}^{(1)}P$$

is true using the first step transition.

We aim to prove $\text{mEngel}(\boldsymbol{d}^{(1)}) = \boldsymbol{d}^{(1)}P$. By the power method, we have $(\boldsymbol{d}^{(1)}P)_i = \sum_{j=1}^{n} d_j^{(1)} p_{ji}$. This means that we need to prove $(\text{mEngel}(\boldsymbol{d}^{(1)}))_i = \sum_{j=1}^{n} d_j^{(1)} p_{ji}$.

The canonical decomposition of P is defined as

$$P = \begin{array}{c} \\ \text{TR.} \\ \text{ABS.} \end{array} \begin{array}{c} \begin{array}{cc} \text{TR.} & \text{ABS.} \end{array} \\ \begin{bmatrix} Q & R \\ \mathbf{0} & I \end{bmatrix} \end{array}$$

where TR. states are transient states, while ABS. states are absorbing states (e.g., see Figure 3). $\mathbf{0}$ represents a zero matrix and I represents an identity matrix with appropriate dimensions.

To do so, let us recall

$$\widehat{P} = \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \\ 1' \\ \vdots \\ n' \end{array} \begin{array}{c} \begin{array}{ccccccc} s & 1 & \cdots & n & 1' & \cdots & n' \end{array} \\ \begin{bmatrix} 0 & d_1^{(1)} & \cdots & d_n^{(1)} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & p_{11} & \cdots & p_{1n} \\ 0 & 0 & \ddots & 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & p_{n1} & \cdots & p_{nn} \\ 0 & 0 & \cdots & 0 & 1 & \cdots & 0 \\ 0 & 0 & \ddots & 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \end{bmatrix} \end{array}$$

Using the absorbing Markov chain formulation, we have

$$Q = \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{cccc} s & 1 & \cdots & n \end{array} \\ \begin{bmatrix} 0 & d_1^{(1)} & \cdots & d_n^{(1)} \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix} \end{array}$$

$$R = \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccc} 1' & \cdots & n' \end{array} \\ \begin{bmatrix} 0 & \cdots & 0 \\ p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \end{array}$$

The fundamental matrix for $\widehat{P}$ is

$$J = (\mathbf{1}_{n+1 \times n+1} - Q)^{-1} = \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{cccc} s & 1 & \cdots & n \end{array} \\ \begin{bmatrix} 1 & d_1^{(1)} & \cdots & d_n^{(1)} \\ 0 & 1 & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & 1 \end{bmatrix} \end{array}$$

and the time-to-absorption matrix is

$$B = JR = \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{cccc} s & 1 & \cdots & n \end{array} \\ \begin{bmatrix} 1 & d_1^{(1)} & \cdots & d_n^{(1)} \\ 0 & 1 & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccc} 1' & \cdots & n' \end{array} \\ \begin{bmatrix} 0 & \cdots & 0 \\ p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \end{array}$$

$$= \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccc} 1' & \cdots & n' \end{array} \\ \begin{bmatrix} \sum_{j=1}^{n} d_j^{(1)} p_{j1} & \cdots & \sum_{j=1}^{n} d_j^{(1)} p_{jn} \\ p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \end{array} = \begin{array}{c} \\ s \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccc} 1' & \cdots & n' \end{array} \\ \begin{bmatrix} d_1^{(2)} & \cdots & d_n^{(2)} \\ p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \end{array}$$

So the $(\text{mEngel}(\boldsymbol{d}^{(1)}))_i$ is B_{1i}, which is $\sum_{j=1}^{n} d_j^{(1)} p_{ji}$. This can also be expressed as $(d^{(1)}P)_i$. From s to $\{1', \cdots, n'\}$ in B, the distribution of $\{1', \cdots, n'\}$ is $\boldsymbol{d}^{(2)}$, i.e.,

$$\text{mEngel}(\boldsymbol{d}^{(1)}) = \boldsymbol{d}^{(2)} = \boldsymbol{d}^{(1)}P.$$

The rest of the iteration results follow, i.e.,

$$\text{mEngel}(\boldsymbol{d}^{(k)}) = \boldsymbol{d}^{(k+1)} = \boldsymbol{d}^{(k)}P,$$

where P can be a transition matrix or an absorbing transition matrix. From $\{1, \cdots, n\}$ to $\{1', \cdots, n'\}$ in B, we check to see that P remains unchanged at each iteration. The $k+1$-step transition will follow.

3.1. Implementation of the mEngel Algorithm

Algorithm 1 presents the pseudo-code structure, whereas Algorithm 2 provides the matrix formulations of $\widehat{N}$ and $\widehat{D}$ for (1) and (2), respectively. Algorithm 3 demonstrates the nested mEngel algorithm with an accuracy tolerance as the stopping criterion. We use N and D for illustrative purposes in these three algorithms to avoid confusion.

Let

$$\boldsymbol{d}^{(0)} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$$

be the initial probability distribution vector. The details of Algorithm 1 are described as follows:

- In Lines 3 - 5, MD represents the capacity array of the states. An element in MD is equal to 0 when it is a recurrent state and not equal to zero when it is a transient state. This part is also called critical loading.
- In Lines 15 - 39, the while loop ends when the initial transient states' chips are the same as the end states' chips.
- In Lines 20 - 30, chips are firing if available. Otherwise, it simply counts the number of unavailable states.
  - In Lines 21 - 22, if the available chips are more than the capacity for chips, chips get stored in $new1$.
  - In Lines 24 - 28, $j \neq i$, j is the order of states that the chips move to, i is the order of states that the chips start from, and j is not equal to i means that in this round of loop, the current state is not the starting state.
- In Lines 31 - 32, if the number of states that are not available is equal to the transient states, the while loop ends.
- In vector $temp$, the total of d is needed, i.e., $sum(d)$, because the starting state needs $sum(d)$ chips to be available to be fired. We set $counteq = 0$, which is the number of unavailable transient states. The loop will stop when $counteq$ is equal to the number of transient states n.
- In Lines 37 - 38, check whether the number of current transient state's chips is the same as the chips of the initial one, i.e., critical loading. If yes, the while loop ends.
- In Lines 40 - 41, the probability distribution vector is updated.
- In Lines 42 - 43, the normalized probability distribution vector is calculated.

The details of Algorithm 2 are described as follows:

- Matrix N of (1): In Lines 1 - 12, matrix N is constructed with the following dimensions: $(2n+1) \times (2n+1)$ where $S = \{s\}$, $\{1, \cdots, n\}$, and $\{1', \cdots, n'\}$ represent the rows and columns:
  - $N[1,1] = 0$
  - $N[1, 2 : n+1] =$ initial distribution $\boldsymbol{d}^{(0)} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$
  - $N[1, n+2 : 2n+1] = 0$
  - The first column is defined as 0.
  - The 2nd through $n+1$ columns are defined as 0.
  - The $n+2$ through $2n+1$ columns are defined as 0.
  - $N[2 : n+1, 1] = 0$
  - The $n \times n$ submatrix $N[n+2 : 2n+1, n+2 : 2n+1]$ is I_n, which is an identity matrix of order n.
  - The remaining elements of N are defined as $\text{ceil}(c \cdot P)$, where P is a transition matrix and c is a constant.
- Matrix D of (2): In Lines 13 - 16, matrix D is constructed with the same dimensions as matrix N.
  - $D[1,1] = m^{(k)}_{1\times 1}$, which is a given scalar value.
  - $D[1, 2 : n+1] = \boldsymbol{m}^{(k)}_{1\times n}$, which is a given scalar vector.
  - $D[1, n+2 : 2n+1] = \boldsymbol{m}^{(k)}_{1\times n}$, which is a given scalar vector.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

- The first column is defined as $c \cdot \mathbf{1}_{n\times 1}$, where c is a constant.
- The 2nd through $n+1$ columns are defined as $c \cdot \mathbf{1}_{n\times 1}$.
- The $n+2$ through $2n+1$ columns are defined as $c \cdot \mathbf{1}_{n\times 1}$.
- The remaining elements of D are defined as $\mathbf{1}_{n\times 1}$ and $\mathbf{1}_{n\times n}$, respectively.

Algorithm 1: Structure of the mEngel algorithm

```
Input: n: the # of states; P: the transient matrix
Output: d: the probability distribution vector
Get N(2n+1, 2n+1) and D(2n+1, 2n+1) using Algorithm 2; d(0 : n) = round(1000 * d(0 : n));
new1(0 : 2n+1) = 0; temp(0 : 2n+1) = 0; MD(0 : 2n+1) = 0; sum1 = 0; x ← TRUE;
for i = 1 to 2n+1 do
    for j = 1 to 2n+1 do
        MD(i) = D(i, j) − 1;
for i = 1 to 2n+1 do
    temp(i) = MD(i);
x = 1;
for i = 2 to n+1 do
    N(1, i) = d(x);
    x = x + 1
for i = 1 to 2n+1 do
    D(1, i) = sum(d);
num = 0;
while x do
    y ← TRUE
    while y do
        z = 0;
        for i = 1 to n+1 do
            if (temp(i) > MD(i) && MD(i) ~= 0) then
                for j = 1 to 2n+1 do
                    new1(j) = temp(j);
                value = temp(i);
                for j = 1 to 2n+1 do
                    if j ~= i then
                        temp(j) = new1(j) + N(i, j);
                    else
                        temp(j) = value − D(i, j);
            else
                z = z + 1;
        if z == n+1 then
            y ← FALSE;
    temp(a) = temp(a) + sum(d); counteq = 0;
    for i = 2 to n+1 do
        if (temp(i) == MD(i) && MD(i) ~= 0) then
            counteq = counteq + 1;
    if (counteq == n && num ~= 0) then
        x ← FALSE;
    num = num + 1;
for i = 1 to n do
    d(i) = temp(i + n + 1);
for i = 1 to n do
    d(i) = d(i)/sum(d);
```

4. Numerical Results

In this section, we apply the mEngel algorithm to the absorbing problem, which is Torrence's problem, and non-absorbing problems such as Monopoly. We use our homemade R code to generate all numerical results and graphical plots. All the transition matrices defined below are input in Algorithms 1 - 3.

Let X_n represent the state at time n, and $P(X_n = j|X_{n-1} = i)$ denote the conditional probability that the state at time $n-1$ is i and at time n is j. Then, we have $P(X_n = j|X_{n-1} = i_{n-1}, \cdots, X_0 = i_0) = P(X_n = j|X_{n-1} = i_{n-1})$, which is known as the Markov property. It is important to note that $P(j|i) = P(X_n = j|X_{n-1} = i)$. For instance, in the Markov matrix M, we have $M_{ij} = P(j|i)$, and it has the property that $\sum_{j\in S} M_{ij} = 1$ for all $i \in S$, where S is the state set.

Algorithm 2: Matrix formulations of N and D

Input: P: the transition matrix; n: the # of states; $c = 10000$
Output: N and D

$S1 \leftarrow round(c * P)$;
$d(0:n) \leftarrow 0$;
$d(1) \leftarrow 1$;
$z \leftarrow 0$;
$z1(0:n) \leftarrow 0$;
$z2(0:2n) \leftarrow 0$;
$z3 \leftarrow matrix((n,n),0)$;
$z4 \leftarrow diag(n)$;
$p1 \leftarrow [z; z2]$;
$p2 \leftarrow [d; z3; z3]$;
$p3 \leftarrow [z1; S1; z4]$;
$N \leftarrow [p1, p2, p3]$;
$q1 \leftarrow matrix((n, 2*n+1), 1)$;
$q2 \leftarrow matrix((n, 2*n+1), c)$;
$q3 \leftarrow matrix((2, n+1), 1)$;
$D \leftarrow [q3; q2; q1]$;

Algorithm 3: Illustration of the nested mEngel algorithm

Input: n: the # of states; d: the distribution of the Markov chain; N: the numerator matrix; D: the denominator matrix
Output: temp/sum1

while k **do**
    $d(0:n) = \mathbf{0}$;
    $d(1) = 1$;
    $k \leftarrow$ TRUE;
    Lines 2 - 60 in Algorithm 1;
    if $sum(abs(d - temp)) < 0.1$ **then**
        $k \leftarrow$ FALSE;
        $d \leftarrow temp$;

4.1. Torrence's problem

Torrence's problem is a random walk problem given as a weekly Riddler puzzle on the FiveThirtyEight website [25]. The transition diagram for Torrence's problem shown in Figure 3 is the one from [22].

The transition matrix of Torrence's problem is given by

$$P = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\ \begin{bmatrix} 0 & 1/3 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 0 & 1/3 & 0 & 0 & 0 & 1/3 & 0 & 0 & 0 \\ 0 & 1/3 & 0 & 1/3 & 0 & 0 & 0 & 1/3 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 1/3 & 0 & 0 & 0 & 1/3 & 0 \\ 1/3 & 0 & 0 & 1/3 & 0 & 0 & 0 & 0 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}.$$

A person begins in the dark inner circle state and aims to reach the other colored states. The probability of each state is shown in Figure 4. The probability of each recurrent state is $5/11$ for the closest state, $2/11$ for the second closest two states, and $1/11$ for the furthest state. These distribution values are consistent with the existing ones. The mEngel's algorithm has 22 configurations, resulting in a stationary distribution mirroring the final round's chip distribution. All transient states eventually hold an equal number of chips to the initial configuration, as shown in Table 1. These findings are consistent with [22].

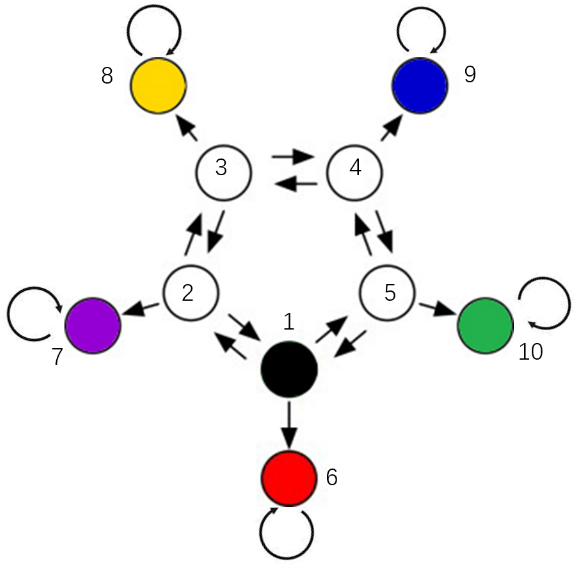

Figure 3. Transition diagram of Torrence's problem

Table 1. Process value distribution of chips for state 1.

Configuration \ State	6	7	8	9	10	1	2	3	4	5	a firing state
0	0	0	0	0	0	2	2	2	2	2	
1	0	0	0	0	0	3	2	2	2	2	0
2	1	0	0	0	0	0	3	2	2	3	1
3	1	1	0	0	0	1	0	3	2	3	2
4	1	1	1	0	0	1	1	0	3	3	3
5	1	1	1	1	0	1	1	1	0	4	4
6	1	1	1	1	1	2	1	1	1	1	5
7	1	1	1	1	1	3	1	1	1	1	0
8	2	1	1	1	1	0	2	1	1	2	1
9	2	1	1	1	1	1	2	1	1	2	0
10	2	1	1	1	1	2	2	1	1	2	0
11	2	1	1	1	1	3	2	1	1	2	0
12	3	1	1	1	1	0	3	1	1	3	1
13	3	2	1	1	1	1	0	2	1	3	2
14	3	2	1	1	2	2	0	2	2	0	5
15	3	2	1	1	2	3	0	2	2	0	0
16	4	2	1	1	2	0	1	2	2	1	1
17	4	2	1	1	2	1	1	2	2	1	0
18	4	2	1	1	2	2	1	2	2	1	0
19	4	2	1	1	2	3	1	2	2	1	0
20	5	2	1	1	2	0	2	2	2	2	1
21	5	2	1	1	2	1	2	2	2	2	0
22	5	2	1	1	2	2	2	2	2	2	0

4.2. Monopoly

Monopoly, a widely enjoyed board game, can be modelled as a non-absorbing Markov chain problem. The game's intricate rules result in a complex transition matrix. We plan to construct this transition matrix by multiplying four separate matrices (e.g, [4]), which we will detail subsequently. We dissect the entire Markov chain into segments, each of which can be considered a sub-chain. Each sub-chain corresponds to a specific rule, and their combination reconstitutes the original chain.

In Monopoly, a player's decision is primarily determined by the ownership of land on the board. In decision-making, the player only needs to consider the probability of landing on each piece of land and the benefits of landing on each piece of land. In this section, we calculated the probability of landing on each piece of land and the benefits of purchasing each piece of land in the two jail strategies.

Using the long and short jail strategies [24], we aim to find the probability of landing in Jail for three scenarios: 1. Landing on "Go to Jail", 2. Drawing a "Go to Jail" card, 3. Rolling doubles three times. Within the non-absorbing Monopoly Markov chains consisting of 41 states, where the first 40

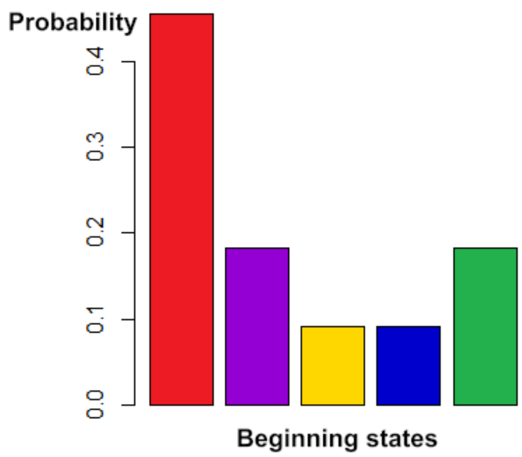

Figure 4. Torrence's distribution

states are regular Monopoly squares as listed, for a short jail strategy, at state 41, if a player starts their turn in Jail, they have the option to play a "Get Out of Jail Free" card (Community Chest or Chance) or pay $50 to roll the dice and exit Jail normally. For a long jail strategy, assuming the player is incarcerated, the non-absorbing Monopoly Markov chains consist of 43 states, where the first 40 states are regular Monopoly squares as listed, state 41 represents the initial turn in Jail, state 42 the second, and state 43 the third. In Monopoly, the player moves around the 40 squares by rolling a pair of fair dice. If the player rolls doubles, they get an extra turn. If they roll doubles again on this extra turn, they get yet another additional turn. However, if the player rolls doubles for a third time, they are sent to Jail instead of proceeding as usual. This is called the 120×120 model [5]. In what follows, we also use long and short jail strategies for obtaining the stationary distribution of the Monopoly Markov chains formulated by the 123×123 model.

4.2.1. Construction of Monopoly Matrices

Firstly, players use two dice to move, with the number of steps corresponding to the sum of the two dice values, as shown in Table 2.

 doi:10.20944/preprints202507.0451.v1

Table 2. Two dice sum probabilities.

Sum of two dice	Probability
2	$P(2) = \frac{1}{36}$
3	$P(3) = \frac{2}{36}$
4	$P(4) = \frac{3}{36}$
5	$P(5) = \frac{4}{36}$
6	$P(6) = \frac{5}{36}$
7	$P(7) = \frac{6}{36}$
8	$P(8) = \frac{5}{36}$
9	$P(9) = \frac{4}{9}$
10	$P(10) = \frac{3}{36}$
11	$P(11) = \frac{2}{36}$
12	$P(12) = \frac{1}{36}$

To end up in Jail, a player must roll 3 doubles to end up in jail, land on the square, or get the card that sends them there. Once in Jail, the player can either pay to leave on their next turn or try to roll doubles. If neither of these options happens, the player must stay in Jail for three turns. These rules are important for understanding the jail matrix and the movement matrix. For example, the movement matrix involves a short jail strategy with an extra state and a long jail strategy with three extra states, in the case of three extra states, 1. If the player chooses not to use a card or pay the fee, they still need to roll the dice. 2. If the player rolls doubles, they move out of Jail the designated number of squares without earning an additional turn, even though doubles are rolled. 3. Failing to roll doubles on the third attempt requires paying $50 to leave and proceed with the indicated number of squares. Landing on a Chance or Community Chest square prompts players to draw a card that could lead them to other squares (the player takes the top card from the draw pile and immediately places it on the discard pile). The probability of drawing a card from the Chance or Community Chest squares is shown in Tables 3 and 4. The "Stay" card groups all non-moving square cases. In Table 3, there are 16 cards available, where 14 of them do not involve moving to another square (they involve either gaining or losing money in some way). Of the 14 "Stay" cards, 9 reward the player with cash, 4 make the player pay money to the bank or other players, and 1 is a "Get Out of Jail Free" card. One card sends the player directly to Jail ("Go to Jail"), and another card sends the player to the start position ("Advance to Go"). Table 4 shows that one card sends the player directly to Jail ("Go to Jail"), three of the six "Stay" cards make the player pay money to the bank or other players, two reward the player with cash, and one is a "Get Out of Jail Free" card. The remaining seven direct the player to a property, railroad, or utility, highlighted in bold.

Table 3. Community Chest Destinations.

Card	Probability
Stay	14/16
Advance to Go	1/16
Go to Jail	1/16

We aim to categorize these rules into four groups, each represented by a matrix. These matrices are named according to the rules they represent: the movement matrix $\boldsymbol{M} = [M_{ij}]$, jail matrix $\boldsymbol{J} = [J_{ij}]$, chance matrix $\boldsymbol{Ch} = [Ch_{ij}]$, and community chest matrix $\boldsymbol{Cc} = [Cc_{ij}]$. All four matrices adhere to an order that mirrors the priority of the rules in Monopoly. For instance, a player must first move and then act based on the new square they land on, making the movement matrix the first. The jail matrix is second as we consider the possibility of jail upon arriving at a new square. The order of the chance and community chest matrices is also fixed. If a player lands on a Chance square, they may be sent to a Community Chest square. This is due to one of the 16 chance cards instructing the player to "Go back 3 spaces", and the Chance square is three ahead of the Community Chest square.

This approach differs from [4], where we attempted to change the order of the chance and community chest matrices, resulting in a different outcome. This paper considers the scenario where a player can move from a Chance square to a Community Chest square. Upon researching the commutative law of matrix multiplication, we found that these two matrices do not satisfy the conditions of the commutative law.

Table 4. Chance Destinations.

Card	Probability
Take a Trip to Reading RailRoad	1/16
Advance to the nearest RailRoad	2/16
Advance to St. Charles Place	1/16
Advance to the Nearest Utility	1/16
Advance to Illinois Avenue	1/16
Advance to Boardwalk	1/16
Advance to Go	1/16
Go Directly to Jail	1/16
Go back three spaces	1/16
Stay	6/16

Upon integrating these matrices, we derive the transition matrix, defined as follows:

$$P = \boldsymbol{MJChCc}.$$

Two strategies emerge: long jail formulated by the 43×43 model and short jail formulated by the 41×41 model. The long jail strategy involves players remaining in Jail for as long as possible, implying they will not pay to leave. Conversely, the short jail strategy permits players to leave immediately on their next turn. The numerical distinction between these two strategies lies in the last three rows of the movement matrix, while the other matrices remain unchanged.

Let us define the movement matrix in the long jail strategy as $\boldsymbol{M}^L$ and the movement matrix in the short jail strategy as $\boldsymbol{M}^S$. The transition matrix in the long jail strategy is given by $P = \boldsymbol{M}^L\boldsymbol{JChCc}$, and the transition matrix in the short jail strategy is given by $P = \boldsymbol{M}^S\boldsymbol{JChCc}$, as shown in Tables A1 and A2 in Appendix A. Next, we will construct the Markov matrix for $\boldsymbol{M}^L, \boldsymbol{M}^S, \boldsymbol{J}, \boldsymbol{Ch}$, and $\boldsymbol{Cc}$ as shown in Tables A3, A4, A5, A6, and A7, respectively, in Appendix A.

The movement matrix in the long jail strategy is given by:

$$M^L_{ij} = \begin{cases} 1/36 & \text{if} \quad j = i+2, i-38, i+12, i-28 \quad \text{and} \quad 1 \le i,j \le 40 \quad \text{or} \quad j = 13, 23, i = 43 \\ & \text{or} \quad j = 13, 15, 17, 19, 21, 23, i = 41, 42 \\ 2/36 & \text{if} \quad j = i+3, i-37, i+11, i-29 \quad \text{and} \quad 1 \le i,j \le 40 \quad \text{or} \quad j = 14, 22, i = 43 \\ 3/36 & \text{if} \quad j = i+4, i-36, i+10, i-30 \quad \text{and} \quad 1 \le i,j \le 40 \quad \text{or} \quad j = 15, 21, i = 43 \\ 4/36 & \text{if} \quad j = i+5, i-35, i+9, i-31 \quad \text{and} \quad 1 \le i,j \le 40 \quad \text{or} \quad j = 16, 20, i = 43 \\ 5/36 & \text{if} \quad j = i+6, i-34, i+8, i-32 \quad \text{and} \quad 1 \le i,j \le 40 \quad \text{or} \quad j = 17, 19, i = 43 \\ 6/36 & \text{if} \quad j = i+7, i-33 \quad \text{and} \quad 1 \le i,j \le 40 \quad \text{or} \quad j = 18, i = 43 \\ 30/36 & \text{if} \quad j = 42, i = 41 \quad \text{or} \quad j = 43, i = 42 \end{cases}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

For the 1st-40th rows: Take the first row as an example. This means that the first row starts from "Go" (1st state) and goes to other states based on the probability of the outcome of the roll of the two dice.

The probability of moving from the ith state to the jth state is $P(j|i) = P(j-i)$. Apply it, and we can get the movement matrix $M^L_{ij} = P(j|i)$. The last three rows have 2 steps:

Step 1 The probability of leaving from Jail is $1/6$, while the probability of staying in Jail is $1 - 1/6 = 5/6$.

Step 2 They have the same formation as the 1st-40th rows if 'leaving Jail'. But in states 41 and 42, the entries M^L_{ij}, (where $i = 41, 42, j < 41$), is $M^L_{ij} = (5/6) = P(j|13)$ ($5/6$ because 'if leaving', from Step 1). And M^L_{ij}, (where $i = 43, j < 41$), is $M^L_{ij} = P(j|13)$.

In Monopoly, the last three states show how to leave Jail. In Jail, in states 41 and 42, players have a $1/36$ chance to roll doubles. If they do, they leave Jail and move forward that many spaces. Because we added extra states for being in Jail, but rolling doubles means they move from the Jail square, if they do roll doubles, they move to state 11, "Just Visiting Jail".

Squares from Jail	11	12	13	14	15	16	17	18	19	20	21	22	23
	11	$11+1$	$11+2$	$11+3$	$11+4$	$11+5$	$11+6$	$11+7$	$11+8$	$11+9$	$11+10$	$11+11$	$11+12$
Probability of Leaving after the First Turn in Jail	0	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36
Probability of Leaving after the Second Turn in Jail	0	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36

A player has a $5/6$ chance of staying in Jail, meaning the chance of moving from state 41 to 42 is $5/6$. This is because state 41 is when they are first put in Jail, and if they do not roll doubles, they move to their first turn in Jail, state 42. The same $5/6$ chance applies to moving from state 42 to 43. After the third turn in Jail, they must pay to leave. So, they leave from state 43 with chances based on the rolling of two dice, like the rest of the matrix. The chances of moving from state 43 to other states on the board are outlined below:

Squares	$1\cdots 11$	12	13	14	15	16	17	18	19	20	21	22	23	$24\cdots 43$
Probability of Leaving after the Third Turn in Jail	$0\cdots 0$	0	1/36	2/36	3/36	4/36	5/36	6/36	5/36	4/36	3/36	2/36	1/36	$0\cdots 0$

The movement matrix in the short jail strategy is given by:

$$M^S_{ij} = \begin{cases} 1/36 & \text{if} \quad j = i+2, i-38, i+12, i-28 \quad \text{and} \quad 1 \le i, j \le 40 \\ & \text{or} \quad j = 13, 23, i = 41, 42, 43 \\ 2/36 & \text{if} \quad j = i+3, i-37, i+11, i-29 \quad \text{and} \quad 1 \le i, j \le 40 \\ & \text{or} \quad j = 14, 22, i = 41, 42, 43 \\ 3/36 & \text{if} \quad j = i+4, i-36, i+10, i-30 \quad \text{and} \quad 1 \le i, j \le 40 \\ & \text{or} \quad j = 15, 21, i = 41, 42, 43 \\ 4/36 & \text{if} \quad j = i+5, i-35, i+9, i-31 \quad \text{and} \quad 1 \le i, j \le 40 \\ & \text{or} \quad j = 16, 20, i = 41, 42, 43 \\ 5/36 & \text{if} \quad j = i+6, i-34, i+8, i-32 \quad \text{and} \quad 1 \le i, j \le 40 \\ & \text{or} \quad j = 17, 19, i = 41, 42, 43 \\ 6/36 & \text{if} \quad j = i+7, i-33 \quad \text{and} \quad 1 \le i, j \le 40 \\ & \text{or} \quad j = 18, i = 41, 42, 43 \end{cases}$$

For the first row and any rows of the 2nd to 40th rows of M^S_{ij}, the probability of moving from the ith state to the jth state is $P(j|i) = P(j-i)$, which is the same as the ones in M^L_{ij}. The last three rows are computed as follows:

- For short jail, the probability of staying in Jail for one more turn is 0. (in comparison with long jail)

- They have the same formation as the 1st-40th rows if 'leaving Jail'. (like M_{ij}^{L})
- Starting with the 11st state (share with "Just Visiting Jail"), the entries $M_{ij}^{S} = P(j|i)$ are the probabilities from rolling two dice, where $j = 13, \cdots, 23$, and $i = 41, 42, 43$.

  The jail matrix is given by

$$J_{ij} = \begin{cases} 215/216 & \text{if} \quad i = j \quad \text{and} \quad 1 \leq i \leq 40 \quad \text{with} \quad i \neq 31 \\ 1/216 & \text{if} \quad j \quad \text{and} \quad 1 \leq i \leq 40 \quad \text{with} \quad i \neq 31 \\ 1 & \text{if} \quad i = 31, j = 41, \quad \text{or} \quad 41 \leq i, j \leq 43 \\ 0 & \text{if} \quad \text{otherwise} \end{cases}$$

where it describes the rule for rolling doubles three times, the probability of which is $(1/6)^3 = 1/216$, so the probability of staying out of Jail is $1 - 1/216 = 215/216$, and $J_{ij} = P(j|i) = 215/216$, when $i = j, j \neq 41$, and $J_{ij} = P(j|i) = 0$, when $i \neq j, j \neq 41$, $J_{ij} = P(j|i) = 1/216$ when $j = 41$.

Special cases: 31st, 41st, 42nd, and 43rd.

- For the 31st, when players land in this state, they will go to Jail. So $J_{ij} = P(j|i) = 0$ when $j \neq 41$, $J_{ij} = 1$ when $j = 41$.
- The 41st, 42nd, and 43rd players who are already in Jail will stay in Jail. So $J_{ij} = P(j|i) = 0$ when $i \neq j$, $J_{ij} = 1$ when $i = j$.

We will now examine the jail matrix. Adding two states and the doubles rule makes it more complex. The chance of rolling a double is $(6/36)^2 = (1/6)(1/6) = 6/36$, so the chance of rolling three doubles in a row is $(6/36)^3$. Each roll is independent, meaning the chance of rolling a double does not change, no matter the previous rolls. So, the chance of getting three doubles in a row is $(6/36)^3 = (1/6)(1/6)(1/6) = (1/216)$. In the 43×43 jail matrix, all the diagonal entries are ones, except for state 31, the Policeman square. Here, the chance to stay in each state is 215/216, and to go to Jail in state 41 is 1/216. Players can be sent to Jail anytime by rolling three doubles in a row. The exception is state 31, where going to Jail is certain. Players in states 41, 42, or 43 stay put. So, we made a jail matrix that accounts for the chance of rolling doubles three times.

The chance matrix is given by

$$Ch_{ij} = \begin{cases} 1/16 & \text{if} \quad j = 1, 6, 12, 25, 40, 41, i = 8, 23, 37, \\ & \qquad [8,6], [8,5], [8,13], [23,20], [23,29], [37,29], [37,34] \\ 2/16 & \text{if} \quad [23,26], [37,36] \\ 3/16 & \text{if} \quad j = 6 \quad \text{and} \quad i = 8 \\ 6/16 & \text{if} \quad j = i \quad \text{and} \quad i = 8, 23, 37 \\ 1 & \text{if} \quad 1 \leq i, j \leq 43 \quad \text{with} \quad 8, 23, 37 \\ 0 & \text{otherwise} \end{cases}$$

The rules for calculating probabilities are as follows:

- For the first and last rows, as well as for all other rows except the 8th, 23rd, and 37th, there is no change. Therefore, $Ch_{ij} = P(j|i) = 0$ for $i \neq j$ when $i \neq 8, 23, 37$.
- Additionally, $Ch_{ij} = P(j|i) = 1$ when $i = j$ and $i \neq 8, 23, 37$.
- In the special case of the 8th, 23rd, and 37th rows, the probability depends on randomly drawn cards. Therefore, there are no specific formulations, and the probability is equal to the probability of the drawn cards as shown in Table 4.

The community chest matrix is given by

$$Cc_{ij} = \begin{cases} 1/16 & \text{if} \quad i = 3, 18, 34 \quad \text{and} \quad 1 \leq j \leq 41 \\ 4/16 & \text{if} \quad i = 3, 18, 34 \quad \text{and} \quad i = j \\ 1 & \text{if} \quad i \neq 3, 18, 34 \quad \text{and} \quad i = j \\ 0 & \text{otherwise} \end{cases}$$

For the 1st and last row (in fact for all rows except the 3rd, 18th, and 34th), there is no change. So, $Cc_{ij} = P(j|i) = 0$ for $i \neq j$, when $i \neq 3, 18, 34$. For $i = j$, when $i \neq 3, 18, 34$, $Cc_{ij} = P(j|i) = 1$. For the 3rd, 18th, and 34th rows, the probability depends on randomly drawn cards, as shown in Table 3.

This matrix has ones down the diagonal except in the 3rd, 18th, and 34th states, where the diagonal is 14/16. This is because 14 cards in the Community Chest will not move the player, and there are two that do-one being "Advance to Go" and the other being "Go to Jail". For the 3rd, 18th, and 34th rows, these cards will send a player to state 1 or state 41, with the probability of each being 1/16.

4.2.2. Numerical Results for the 43 × 43 Model (Long Jail) and for the 41 × 41 Model (Short Jail)

Solutions for the stationary distribution of the Monopoly Markov chains using the mEngel algorithm, the power method, and the canonical decomposition of absorbing Markov chains are shown in Table 5. There is no sizable difference in the results for all the distribution values when they are judged to three decimal places. A comparison of the convergence of the Monopoly Markov chains using the mEngel algorithm and the power method is shown in Table 6. These results indicated that by reducing the tolerance, adjusting the parameters such as c, and increasing the number of iterations, the differences between the two results get smaller and smaller, which implies that the solutions for the mEngel algorithm are close to the results for the power method. Ranking based on the stationary distribution of the Monopoly Markov chains using the long jail strategy and the short jail strategy is shown in Table 7. The ranking order of our mEngel results is the same as [1], [6] and [4]. The probabilities of Jail using the 43 × 43 model and the 41 × 41 model as shown in Table 8 are 11.70% and 6.302%, respectively. The most landed on the square in both strategies is Jail. Following "In Jail/Just Visiting", the second most frequently visited square on the board is Illinois Avenue. Note that there exists an Advance to Illinois Avenue card in the Chance deck. In terms of the short jail strategy, the subsequent most commonly landed-on squares are St. James Place, Tennessee Avenue and New York Avenue, situated 6, 8, and 9 squares beyond Jail, respectively. Based on the results of these two strategies, the player should aim to get out of Jail as quickly as possible by paying the fine before all properties have been purchased.

doi:10.20944/preprints202507.0451.v1

Table 5. Solutions for the stationary distribution of the Monopoly Markov chains using the mEngel algorithm, the power method, and the canonical decomposition of absorbing Markov chains.

	mEngel		**Power**		**Canonical form**	
States	Long Jail	Short Jail	Long Jail	Short Jail	Long Jail	Short Jail
Go	2.9220E-02	3.1050E-02	2.9225E-02	3.1048E-02	2.9225E-02	3.1048E-02
Mediterranean Avenue	2.0355E-02	2.1629E-02	2.0360E-02	2.1622E-02	2.0360E-02	2.1622E-02
Community Chest (South)	1.8052E-02	1.9184E-02	1.8056E-02	1.9172E-02	1.8056E-02	1.9172E-02
Baltic Avenue	2.0722E-02	2.2016E-02	2.0725E-02	2.2004E-02	2.0725E-02	2.2004E-02
Income Tax	2.2233E-02	2.3617E-02	2.2238E-02	2.3607E-02	2.2238E-02	2.3607E-02
Reading RailRoad	2.8312E-02	2.9964E-02	2.8316E-02	2.9953E-02	2.8316E-02	2.9953E-02
Oriental Avenue	2.1544E-02	2.2874E-02	2.1544E-02	2.2863E-02	2.1544E-02	2.2863E-02
Chance (South)	8.2434E-03	8.7516E-03	8.2425E-03	8.7468E-03	8.2425E-03	8.7468E-03
Vermont Avenue	2.2046E-02	2.3404E-02	2.2042E-02	2.3388E-02	2.2042E-02	2.3388E-02
Connecticut Avenue	2.1919E-02	2.3266E-02	2.1911E-02	2.3248E-02	2.1911E-02	2.3248E-02
Just Visiting	2.1630E-02	2.2954E-02	2.1617E-02	2.2934E-02	2.1617E-02	2.2934E-02
St. Charles Place	2.5807E-02	2.7281E-02	2.5792E-02	2.7262E-02	2.5792E-02	2.7262E-02
Electric Company	2.5050E-02	2.4882E-02	2.5036E-02	2.4864E-02	2.5036E-02	2.4864E-02
States Avenue	2.2000E-02	2.4003E-02	2.1982E-02	2.3990E-02	2.1982E-02	2.3990E-02
Virginia Avenue	2.4455E-02	2.4864E-02	2.4438E-02	2.4855E-02	2.4438E-02	2.4855E-02
Pennsylvania RailRoad	2.3798E-02	2.6501E-02	2.3781E-02	2.6498E-02	2.3781E-02	2.6498E-02
St. James Place	2.6921E-02	2.8063E-02	2.6909E-02	2.8065E-02	2.6909E-02	2.8065E-02
Community Chest (West)	2.2957E-02	2.5970E-02	2.2949E-02	2.5976E-02	2.2949E-02	2.5976E-02
Tennessee Avenue	2.8093E-02	2.9245E-02	2.8087E-02	2.9254E-02	2.8087E-02	2.9254E-02
New York Avenue	2.7827E-02	3.0561E-02	2.7822E-02	3.0575E-02	2.7822E-02	3.0575E-02
Free Parking	2.7947E-02	2.8516E-02	2.7949E-02	2.8531E-02	2.7949E-02	2.8531E-02
Kentucky Avenue	2.5740E-02	2.7947E-02	2.5742E-02	2.7963E-02	2.5742E-02	2.7963E-02
Chance (North)	1.0270E-02	1.0290E-02	1.0272E-02	1.0296E-02	1.0272E-02	1.0296E-02
Indiana Avenue	2.5216E-02	2.6884E-02	2.5219E-02	2.6900E-02	2.5219E-02	2.6900E-02
Illinois Avenue	2.9513E-02	3.1416E-02	2.9519E-02	3.1432E-02	2.9519E-02	3.1432E-02
B & O RailRoad	2.8496E-02	3.0215E-02	2.8502E-02	3.0231E-02	2.8502E-02	3.0231E-02
Atlantic Avenue	2.5031E-02	2.6705E-02	2.5033E-02	2.6717E-02	2.5033E-02	2.6717E-02
Ventnor Avenue	2.4844E-02	2.6434E-02	2.4846E-02	2.6444E-02	2.4846E-02	2.6444E-02
Water Works	2.7545E-02	2.9165E-02	2.7547E-02	2.9175E-02	2.7547E-02	2.9175E-02
Marvin Gardens	2.4035E-02	2.5507E-02	2.4036E-02	2.5512E-02	2.4036E-02	2.5512E-02
Go to Jail	0	0	0	0	0	0
Pacific Avenue	2.4926E-02	2.6460E-02	2.4927E-02	2.6461E-02	2.4927E-02	2.6461E-02
North Carolina Avenue	2.4466E-02	2.5974E-02	2.4469E-02	2.5974E-02	2.4469E-02	2.5974E-02
Community Chest (East)	2.2259E-02	2.3655E-02	2.2262E-02	2.3657E-02	2.2262E-02	2.3657E-02
Pennsylvania Avenue	2.3381E-02	2.4836E-02	2.3384E-02	2.4840E-02	2.3384E-02	2.4840E-02
Short Line RailRoad	2.5532E-02	2.7122E-02	2.5536E-02	2.7130E-02	2.5536E-02	2.7130E-02
Chance (East)	8.1276E-03	8.6306E-03	8.1287E-03	8.6337E-03	8.1287E-03	8.6337E-03
Park Place	2.0597E-02	2.1869E-02	2.0601E-02	2.1877E-02	2.0601E-02	2.1877E-02
Luxury Tax	2.0564E-02	2.1834E-02	2.0566E-02	2.1839E-02	2.0566E-02	2.1839E-02
Boardwalk	2.4946E-02	2.6390E-02	2.4951E-02	2.6391E-02	2.4951E-02	2.6391E-02
Jail (First turn)	3.7755E-02	4.0069E-02	3.7755E-02	4.0073E-02	3.7755E-02	4.0073E-02
Jail (Second turn)	3.1430E-02	0	3.1462E-02	0	3.1462E-02	0
Jail (Third turn)	2.6192E-02	0	2.6219E-02	0	2.6219E-02	0

 doi:10.20944/preprints202507.0451.v1

Table 6. Comparison of the convergence of the Monopoly Markov chains using the mEngel algorithm and the power method.

Tolerance \ $\| d_{i,\text{mEngel}} - d_{i,\text{Power}} \|_2$	Number of iterations	Long Jail	Short Jail
0.0001	1	1.2969E-05	1.2969E-05
0.00001	2	8.2213E-06	8.2733E-06
0.001	3	1.6723E-04	1.7547E-04
0.001	4	6.2985E-04	7.6839E-04
0.001 (0.01)	5	9.8909E-04	1.1038E-03
0.01	6	1.3026E-03	1.5551E-03
0.01	7	1.4098E-03	1.8264E-03
0.01	8	1.3658E-03	1.8486E-03
0.01	9	1.3059E-03	2.1228E-03
0.01	10	1.3650E-03	2.3236E-03
0.001 (0.01)	11	9.6764E-04	2.1712E-03
0.001 (0.01)	12	9.0800E-04	1.8861E-03
0.001 (0.01)	13	7.6921E-04	1.6592E-03
0.001 (0.01)	14	5.6854E-04	1.4104E-03
0.001 (0.01)	15	6.2546E-04	1.3701E-03
0.001 (0.01)	16	4.5819E-04	1.4093E-03
0.001 (0.01)	17	3.1808E-04	1.2781E-03
0.001	18	2.4727E-04	8.9537E-04
0.001	19	1.3592E-04	6.5633E-04
0.001	20	1.1921E-04	6.0545E-04
0.0001 (0.001)	21	9.8218E-05	4.3479E-04
0.001	22	1.0943E-04	3.1397E-04
0.0001 (0.001)	23	8.0613E-05	2.4965E-04
0.0001 (0.001)	24	5.5567E-05	2.4997E-04
0.0001 (0.001)	25	3.5889E-05	1.5996E-04
0.0001 (0.001)	26	5.0507E-05	1.8358E-04
0.0001 (0.001)	27	7.1482E-05	2.5761E-04
0.0001 (0.001)	28	7.8835E-05	2.4739E-04
0.0001 (0.001)	29	3.1048E-05	1.7960E-04
0.0001	30	1.7391E-05	8.6980E-05

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

Table 7. Ranking based on the stationary distribution of the Monopoly Markov chains using the long jail strategy for the 43 × 43 model and the short jail strategy for the 41 × 41 model.

Rank	States	Long Jail	States	Short Jail
1	Jail (First turn)	3.7755E-02	Jail (First turn)	4.0069E-02
2	Jail (Second turn)	3.1430E-02	Illinois Avenue	3.1416E-02
3	Illinois Avenue	2.9513E-02	Go	3.1050E-02
4	Go	2.9220E-02	New York Avenue	3.0561E-02
5	B & O RailRoad	2.8496E-02	B & O RailRoad	3.0215E-02
6	Reading RailRoad	2.8312E-02	Reading RailRoad	2.9964E-02
7	Tennessee Avenue	2.8094E-02	Tennessee Avenue	2.9245E-02
8	Free Parking	2.7947E-02	Water Works	2.9165E-02
9	New York Avenue	2.7827E-02	Free Parking	2.8516E-02
10	Water Works	2.7545E-02	St. James Place	2.8063E-02
11	St. James Place	2.6921E-02	Kentucky Avenue	2.7948E-02
12	Jail (Third turn)	2.6192E-02	St. Charles Place	2.7281E-02
13	St. Charles Place	2.5807E-02	Short Line RailRoad	2.7122E-02
14	Kentucky Avenue	2.5740E-02	Indiana Avenue	2.6885E-02
15	Short Line RailRoad	2.5533E-02	Atlantic Avenue	2.6705E-02
16	Indiana Avenue	2.5216E-02	Pennsylvania RailRoad	2.6501E-02
17	Electric Company	2.5050E-02	Pacific Avenue	2.6460E-02
18	Atlantic Avenue	2.5031E-02	Ventnor Avenue	2.6435E-02
19	Boardwalk	2.4946E-02	Boardwalk	2.6387E-02
20	Pacific Avenue	2.4926E-02	North Carolina Avenue	2.5974E-02
21	Ventnor Avenue	2.4844E-02	Community Chest (West)	2.5970E-02
22	North Carolina Avenue	2.4466E-02	Marvin Gardens	2.5507E-02
23	Virginia Avenue	2.4455E-02	Electric Company	2.4881E-02
24	Marvin Gardens	2.4035E-02	Virginia Avenue	2.4864E-02
25	Pennsylvania RailRoad	2.3798E-02	Pennsylvania Avenue	2.4837E-02
26	Pennsylvania Avenue	2.3381E-02	States Avenue	2.4003E-02
27	Community Chest (West)	2.2957E-02	Community Chest (East)	2.3655E-02
28	Community Chest (East)	2.2259E-02	Income Tax	2.3618E-02
29	Income Tax	2.2233E-02	Vermont Avenue	2.3404E-02
30	Vermont Avenue	2.2046E-02	Connecticut Avenue	2.3266E-02
31	States Avenue	2.1999E-02	Just Visiting	2.2954E-02
32	Connecticut Avenue	2.1919E-02	Oriental Avenue	2.2874E-02
33	Just Visiting	2.1630E-02	Baltic Avenue	2.2016E-02
34	Oriental Avenue	2.1544E-02	Park Place	2.1869E-02
35	Baltic Avenue	2.0722E-02	Luxury Tax	2.1834E-02
36	Park Place	2.0598E-02	Mediterranean Avenue	2.1629E-02
37	Luxury Tax	2.0564E-02	Community Chest (South)	1.9181E-02
38	Mediterranean Avenue	2.0355E-02	Chance (North)	1.0290E-02
39	Community Chest (South)	1.8052E-02	Chance (South)	8.7516E-03
40	Chance (North)	1.0270E-02	Chance (East)	8.6306E-03
41	Chance (South)	8.2434E-03	Go to Jail	0
42	Chance (East)	8.1276E-03	Jail (Second turn)	0
43	Go to Jail	0	Jail (Third turn)	0

Table 8. The probabilities of landing on Jail using the long jail strategy and the short jail strategy for the 43 × 43 model, the 41 × 41 model, and the 123 × 123 model.

Rank	States	Long Jail 43 × 43 model	Rank	States	Short Jail 41 × 41 model
1	Jail (First turn)	3.7755E-02	1	Jail (First turn)	4.0069E-02
2	Jail (Second turn)	3.1430E-02	31	Just Visiting	2.2954E-02
12	Jail (Third turn)	2.6192E-02	43	Go to Jail	0
33	Just Visiting	2.1630E-02			
43	Go to Jail	0			
		1.1701E-01			**6.3024E-02**
Rank	States	Long Jail 123 × 123 model	Rank	States	Short Jail 123 × 123 model
1	Jail	8.3424E-02	1	Jail	3.5222E-02
28	Just Visiting	2.2933E-02	30	Just Visiting	2.3671E-02
41	Go to Jail	0	41	Go to Jail	0
		1.0636E-01			**5.8892E-02**

4.3. *The 123 × 123 Model*

Let us define $\widehat{S}$ as the order set of 123 states and S as the order set of 43 states. Two sets $\widehat{S}$ and S have the same order as their matrices. And we have divided each state s_i (except Jail) into 3 states s_i^1, s_i^2, s_i^3. For

$$S = \{s_1, s_2, \cdots, s_{40}, j_1, j_2, j_3\},$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

then

$$\widehat{S} = \{s_1^1, s_2^1, \cdots, s_{40}^1, s_1^2, s_2^2, \cdots, s_{40}^2, s_1^3, s_2^3, \cdots, s_{40}^3, j_1, j_2, j_3\}.$$

Let us construct the transition matrix $\widehat{P}$ for the 123×123 model by making use of the first 40 squares of the regular transition matrix

$$P = \boldsymbol{M}_{43\times43}\widehat{\boldsymbol{J}}_{43\times43}\boldsymbol{Ch}_{43\times43}\boldsymbol{Cc}_{43\times43}$$

and the last three states in the long jail strategy, where $\boldsymbol{M}$, $\boldsymbol{Ch}$, and $\boldsymbol{Cc}$ are same as for the 43×43 model, and $\widehat{\boldsymbol{J}}$ is different from $\boldsymbol{J}$. $\widehat{\boldsymbol{J}}$ is a diagonal matrix, but in the 31st row $\widehat{\boldsymbol{J}}_{31,31} = 0$ and $\widehat{\boldsymbol{J}}_{31,41} = 1$, which is "Go to Jail":

$$\widehat{P} = \begin{array}{c} \\ 1-40 \\ 41-80 \\ 81-120 \\ 121-123 \end{array} \begin{array}{c} \begin{array}{cccc} 1-40 & 41-80 & 81-120 & 121-123 \end{array} \\ \begin{bmatrix} (\frac{5}{6})P & (\frac{1}{6})(\frac{5}{6})P & (\frac{1}{6})(\frac{1}{6})P & J1 \\ (\frac{5}{6})P & (\frac{1}{6})(\frac{5}{6})P & (\frac{1}{6})(\frac{1}{6})P & J1 \\ (\frac{5}{6})P & (\frac{1}{6})(\frac{5}{6})P & (\frac{1}{6})(\frac{1}{6})P & J1 \\ J3 & 0 & 0 & J2 \end{bmatrix} \end{array},$$

where $J1_{ij} = \begin{cases} P_{(i \ mod \ 40)41} & j = 121 \\ 0 & j = 122, 123 \end{cases}$, $J2_{ij} = P_{(i-80)(j-80)}$, and $J3_{ij} = P_{(i-80)j}$.

To clarify, $J1$ represents the transition probability from 40 normal states to 3 Jail states. Each state is equivalent to the state with an index 40 greater and 80 greater when used as the starting state for this round. Therefore, the probability $\widehat{P}$ in $[1-40, 121-123]$, which denotes the probability of moving from the '1st to the 40th' state to the '121st to the 123rd' state,, should be the same as $\widehat{P}$ in $[41-80, 121-123]$ and $[81-120, 121-123]$. As a result, $[1-40, 121-123]$, $[41-80, 121-123]$, and $[81-120, 121-123]$ are all denoted as $J1$. Moreover, the transition from $[121-123]$ to $[1-40]$ represents starting from 3 Jail states and transitioning to 40 normal states, which is equivalent to $[41-43, 1-40]$ in P. Thus, this part should be labelled as $J3$. The transition from $[121-123]$ to $[41-120]$ represents starting from 3 Jail states and transitioning to 40 normal states with some doubles. However, since players cannot roll doubles to go to Jail when 'leaving Jail', the probability for this transition is 0. The transition from 3 Jail states to 3 Jail states is labelled as $J2$ because it is equivalent to $[41-43, 41-43]$ in P.

If we use the transition matrix in the long jail strategy to calculate P, we will obtain the transition matrix for the long jail strategy with 120 states. Similarly, if we use the transition matrix in the short jail strategy to calculate P, we will obtain the transition matrix for the short jail strategy with 120 states.

It is important to note that both $\widehat{P}_{ij}$ and P_{ij} represent the probability of going from state i to state j. The difference lies in the order of states, and $\widehat{P}$ has more states than P, as explained above (see S and $\widehat{S}$). Understanding the 123×123 model is crucial for determining the relationship between the i-th state in P and $\widehat{P}$. In $\widehat{P}_{ij}$, i is the starting state, and j is the ending state, using the symbols in S and $\widehat{S}$. When s_i is the starting state, it is the same as s_i^1, s_i^2, and s_i^3, and this implies $P_{x|s_i} = \widehat{P}_{x|s_i^1} = \widehat{P}_{x|s_i^2} = \widehat{P}_{x|s_i^3}$. However, when s_i is the ending state, it is not the same as s_i^1, s_i^2, and s_i^3, but follows the relationship $P_{si|x} = (\frac{5}{6})\widehat{P}_{s_i^1|x} = (\frac{1}{6})(\frac{5}{6})\widehat{P}_{s_i^1|x} = (\frac{1}{6})(\frac{1}{6})\widehat{P}_{s_i^1|x}$, where $x \in \widehat{S}$.

The structure consists of 43 states, including 40 regular states and 3 Jail states, and 123 states including 120 regular states and 3 Jail states. No additional jail states have been added.

Ranking based on the stationary distribution of the Monopoly Markov chains using the long jail strategy and the short jail strategy for the 123×123 model is shown in Table 9. These two ranking results are in similar order. The probabilities of landing on Jail using the 123×123 model as shown in Table 8 are 10.64% and 5.89%, respectively. The most landed on the square in both strategies is Jail. Following "In Jail/Just Visiting", the subsequent frequently visited square on the board is Illinois Avenue for long jail, but Illinois Avenue is the third place for short jail.

Table 9. Ranking based on the stationary distribution of the Monopoly Markov chains using the long jail strategy and the short jail strategy for the 123 × 123 model.

Rank	States	Long Jail	States	Short Jail
1	Jail	8.3424E-02	Jail	3.5222E-02
2	Illinois Avenue	2.9743E-02	Go	3.1326E-02
3	Go	2.9160E-02	Illinois Avenue	3.1233E-02
4	Reading RailRoad	2.9144E-02	Reading RailRoad	3.0606E-02
5	Tennessee Avenue	2.8775E-02	New York Avenue	3.0448E-02
6	New York Avenue	2.8685E-02	B & O RailRoad	2.9929E-02
7	B & O RailRoad	2.8552E-02	Tennessee Avenue	2.9115E-02
8	Free Parking	2.8474E-02	Water Works	2.8870E-02
9	St. James Place	2.7770E-02	Free Parking	2.8416E-02
10	St. Charles Place	2.7218E-02	St. James Place	2.8065E-02
11	Water Works	2.7150E-02	St. Charles Place	2.8034E-02
12	Kentucky Avenue	2.6364E-02	Kentucky Avenue	2.7836E-02
13	Electric Company	2.6236E-02	Short Line RailRoad	2.7040E-02
14	Indiana Avenue	2.5571E-02	Indiana Avenue	2.6761E-02
15	Virginia Avenue	2.5452E-02	Pennsylvania RailRoad	2.6662E-02
16	Pennsylvania RailRoad	2.4969E-02	Boardwalk	2.6606E-02
17	Atlantic Avenue	2.4905E-02	Atlantic Avenue	2.6411E-02
18	Short Line RailRoad	2.4834E-02	Pacific Avenue	2.6226E-02
19	Boardwalk	2.4734E-02	Ventnor Avenue	2.6132E-02
20	Ventnor Avenue	2.4582E-02	Community Chest (West)	2.5814E-02
21	Pacific Avenue	2.4270E-02	North Carolina Avenue	2.5762E-02
22	Community Chest (West)	2.3791E-02	Electric Company	2.5514E-02
23	North Carolina Avenue	2.3776E-02	Marvin Gardens	2.5245E-02
24	Marvin Gardens	2.3557E-02	Virginia Avenue	2.5165E-02
25	States Avenue	2.3298E-02	Pennsylvania Avenue	2.4701E-02
26	Vermont Avenue	2.3168E-02	States Avenue	2.4455E-02
27	Connecticut Avenue	2.3137E-02	Income Tax	2.4130E-02
28	Just Visiting	2.2933E-02	Vermont Avenue	2.4088E-02
29	Income Tax	2.2788E-02	Connecticut Avenue	2.3972E-02
30	Pennsylvania Avenue	2.2704E-02	Just Visiting	2.3671E-02
31	Oriental Avenue	2.2373E-02	Community Chest (East)	2.3498E-02
32	Community Chest (East)	2.1616E-02	Oriental Avenue	2.3464E-02
33	Baltic Avenue	2.1038E-02	Baltic Avenue	2.2425E-02
34	Mediterranean Avenue	2.0362E-02	Luxury Tax	2.1955E-02
35	Luxury Tax	2.0201E-02	Park Place	2.1925E-02
36	Park Place	2.0149E-02	Mediterranean Avenue	2.1925E-02
37	Community Chest (South)	1.8191E-02	Community Chest (South)	1.9492E-02
38	Chance (North)	1.0369E-02	Chance (North)	1.0243E-02
39	Chance (South)	8.6136E-03	Chance (South)	8.9927E-03
40	Chance (East)	7.9217E-03	Chance (East)	8.6250E-03
41	Go to Jail	0	Go to Jail	0

4.4. The Return of Monopoly

We know that some properties (states) in Monopoly have the same colours. If we collect all states with the same colour, we will profit from it. Return is the ratio of expected money and cost. It shows how much money a player can take back at every turn. Players should buy all the properties of one colour and choose the colour based on which will provide the highest returns.

The formula for determining the return based on the number of turns is given by [16]:

$$\text{Turn} = \frac{\text{Cost}}{p \cdot R \cdot E(x)}$$

where Cost is the development cost for a house or a hotel and is different for each property, p is the probability of landing on the property, R is the rent earnings, and $E(x)$ is the expectation of rolling x times in one turn. If a player rolls doubles three times, this player will go to Jail. So, a player can roll doubles at most three times before the next player rolls. Let x be the number of rolls in one turn, $x \in \{1, 2, 3\}$. Then $p(x)$ is the probability of a player rolling x times in one turn. Then $p(1) = \frac{30}{36} = \frac{5}{6}$, where the player rolls no doubles. $p(2) = \frac{6}{36}\left(\frac{30}{36}\right) = \frac{1}{6}\left(\frac{5}{6}\right)$, where the player rolls doubles once and non-doubles once. And $p(3) = \frac{6}{36}\left(\frac{6}{36}\right)(1) = \frac{1}{6}\left(\frac{1}{6}\right)$, where the player rolls two doubles and then, regardless of what the player rolls next, it will end his turn. So the expectation of rolling x times in one turn is:

$$E(x) = 1 \cdot p(1) + 2 \cdot p(2) + 3 \cdot p(3) = 1.19\overline{4}.$$

 doi:10.20944/preprints202507.0451.v1

The rent and cost values are sourced from [12,24]. Let us consider Mediterranean Avenue as an example, using the long jail strategy shown in Table A8 (See Appendix B). If the player owns Mediterranean Avenue, it takes approximately 1234 turns to recoup the cost of the purchase price. Similarly, if the player builds a house, generating revenue will take approximately 699 turns. Here are the rest of the cases:

- For $p = 0.020355$, Cost $= 60$, $R = 2$, the number of the turns is calculated as Turn $= 60 \cdot (0.02)^{-1} \cdot 2^{-1} \cdot (1.19\overline{4})^{-1} = 1233.914$.
- For 1 house, with $R = 10$ and Cost $= 170$, # of Turns $= 170 \cdot (0.020355)^{-1} \cdot 10^{-1} \cdot (1.19\overline{4})^{-1} = 699.2179$.
- For 2 houses, with $R = 30$ and Cost $= 220$, # of Turns $= 220 \cdot (0.020355)^{-1} \cdot 30^{-1} \cdot (1.19\overline{4})^{-1} = 301.6234$.
- For 3 houses, with $R = 90$ and Cost $= 270$, # of Turns $= 270 \cdot (0.020355)^{-1} \cdot 90^{-1} \cdot (1.19\overline{4})^{-1} = 123.3914$.
- For 4 houses, with $R = 160$ and Cost $= 320$, # of Turns $= 320 \cdot (0.020355)^{-1} \cdot 160^{-1} \cdot (1.19\overline{4})^{-1} = 82.2609$.
- For a hotel, with $R = 250$ and Cost $= 370$, # of Turns $= 370 \cdot (0.020355)^{-1} \cdot 250^{-1} \cdot (1.19\overline{4})^{-1} = 60.8731$.

Now, let us examine the example of Mediterranean Avenue using the long jail strategy for the 123×123 model, as shown in Table A9 (see Appendix B). An inspection of Table A9 reveals that the number of turns required to recoup the purchase price of Mediterranean Avenue, including the player's properties, houses, and a hotel, is nearly identical to that of the 43×43 model.

Similar calculations were performed for all the states to calculate the number of turns in Monopoly needed to recoup the cost of the purchase price for each property. This was done using the long jail strategy for both the 43×43 model and the 123×123 model, and using the short jail strategy for both the 41×41 model and the 123×123 model. The results are summarized in Tables A8, A9, A10, and A11 in Appendix B, respectively, where NA stands for not applicable. Although the stationary probabilities varied slightly for each property, the number of turns remained the same. In Tables A8, A9, A10, and A11 in Appendix B, we also observe how fast a player can start to make a profit from each state and each colour group. For the state, Boardwalk and New York Avenue require the least number of turns before they start making money.

The rule for building on properties of the same colour in Monopoly is that a player must own all the sites of the same colour before building on them. We take Brown (Mediterranean Avenue and Baltic Avenue) as an example. To build one house on Mediterranean Avenue, they must first buy all the brown land. So, a player must pay for all the land before building a house. The land cost is 60 for each. Then the house cost is 50 for each. So, the cost of land on brown is $2(60) = 120$, the cost to build 1 house on brown is $2(60 + 50) = 220$, and the cost to build 2 houses on brown is $2(60 + (50)2) = 320$. Therefore the cost of building 1 house on Mediterranean Avenue is $2(60) + 50 = 170$. We want to calculate how much we need to pay to build h houses in some states. Let baseCost be the cost of buying all lands of the same colour, and houseCost be the cost of building one house in a state. We first need to buy all the land of the same colour to build a house. So we need to pay baseCost first, and then we pay $h \times$ houseCost on building h houses. Hence we cost baseCost $+ h \times$ houseCost to build h houses on the state, which is noted as Cost $=$ baseCost $+ h \times$ houseCost. Let n be the number of states in some colour. If we want to calculate the cost of building h houses on each site in the same colour group, which is noted as $\text{Cost}_{\text{color}}$, we need to pay baseCost to buy all lands and pay $h \times n \times$ houseCost on building houses. So $\text{Cost}_{\text{color}} = \text{baseCost} + h \times n \times \text{houseCost}$. We have $h \times \text{houseCost} = \text{Cost} - \text{baseCost}$ then $\text{Cost}_{\text{color}} = \text{baseCost} + n \times (\text{Cost} - \text{baseCost})$. For properties of the same colour group, the formula is given as:

$$\text{Turn} = \frac{\text{Cost}_{\text{color}}}{\sum p \cdot \text{Average of Rent} \cdot E(x)}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

where $\text{Cost}_{\text{color}} = \text{Cost} \cdot n - \text{baseCost} \cdot (n-1)$.

For instance, let us consider the Brown property using the short jail strategy. Below is the number of turns required for the Brown property group to generate income, including the player's properties and 2 houses. Similar calculations apply to other houses and a hotel:

- $p = 0.02125 + 0.02158 = 0.04282$, baseCost = 60 + 60 = 120, $R = \frac{2+4}{2} = 3$, the number of the turns is calculated as # of Turns $= 120 \cdot (0.04282)^{-1} \cdot 3^{-1} \cdot (1.19\bar{4})^{-1} = 782.0731$.
- For 2 houses, with $R = \frac{30+60}{2} = 45$, cost $= 220 + 220 - \text{baseCost} \cdot (2-1) = 440$, # of Turns $= 320 \cdot (0.04282)^{-1} \cdot 45^{-1} \cdot (1.19\bar{4})^{-1} = 307.0004$.

Table 10 summarises similar calculations for all color groups using the long jail strategy for both the 43×43 model and 123×123 model, and using the short jail strategy for both the 41×41 model and the 123×123 model, respectively. The table shows the number of turns it takes for a player to recoup the cost of the purchase price. We compared the ranking based on the number of turns in Monopoly required to recoup the costs of owning hotels for the same color groups. Our observations indicate that the Orange and Light Blue properties are the quickest to start making money of all the colors.

The hotel returns ranking findings for both strategies are in the same order and are consistent with [16] and [7].

doi:10.20944/preprints202507.0451.v1

 doi:10.20944/preprints202507.0451.v1

Table 10. Turns in Monopoly for the same colour property using the long jail strategy for both the 43 × 43 model and the 123 × 123 model, and using the short jail strategy for both the 41 × 41 model and the 123 × 123 model from top to bottom.

Same color property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Brown	4.108E-02	3	120	815.1989	15	220	298.9063	45	320	144.9243	135	420	63.40436	240	520	44.15661	350	620	36.10167
Light Blue	6.551E-02	6.67	320	613.1272	33.33	470	180.2142	93.33	620	84.89788	280	770	35.14464	416.67	920	28.21773	566.67	1070	24.13126
Light Purple	7.226E-02	10.67	440	477.776	53.33	740	160.7667	160	1040	75.30944	466.67	1340	33.26833	650	1640	29.23254	800	1940	28.09622
Orange	8.284E-02	14.67	560	385.7908	73.33	860	118.5252	206.67	1160	56.725	566.67	1260	22.47161	766.67	1560	20.56412	966.67	1860	19.44593
Red	8.047E-02	18.67	680	378.9349	93.33	1130	125.9671	266.67	1580	61.64289	716.67	2030	29.46978	891.67	2480	28.9366	1066.67	2930	28.57838
Yellow	7.391E-02	22.66	800	399.9088	113.33	1250	124.9384	340	1700	56.63708	816.67	2150	29.82103	991.67	2600	29.69867	1166.67	3050	29.61302
Green	7.277E-02	20	920	529.224	136.67	1520	127.9535	410	2120	59.48859	933.33	2720	33.52859	1133.33	3320	33.7026	1316.67	3920	34.25238
Dark Blue	4.554E-02	42.5	750	324.4243	187.5	1150	112.7555	550	1550	51.80957	1250	1950	28.67911	1500	2350	28.80167	1750	2750	28.88921
RailRoad	1.061E-01	200	800	31.5511															
Utility	5.260E-02	70	300	68.2137															

Same color property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Brown	4.140E-02	3	120	808.8979	15	220	296.5959	45	320	143.8041	135	420	62.91428	240	520	43.8153	350	620	35.82262
Light Blue	6.868E-02	6.67	320	584.8277	33.33	470	171.8962	93.33	620	80.97933	280	770	33.5225	416.67	920	26.91531	566.67	1070	23.01746
Light Purple	7.597E-02	10.67	440	454.4438	53.33	740	152.9157	160	1040	71.6317	466.67	1340	31.64367	650	1640	27.80497	800	1940	26.72414
Orange	8.523E-02	14.67	560	374.9725	73.33	860	115.2016	206.67	1160	55.13433	566.67	1260	21.84147	766.67	1560	19.98746	966.67	1860	18.90063
Red	8.168E-02	18.67	680	373.3214	93.33	1130	124.1011	266.67	1580	60.72971	716.67	2030	29.03321	891.67	2480	28.50794	1066.67	2930	28.15502
Yellow	7.304E-02	22.66	800	404.6722	113.33	1250	126.4266	340	1700	57.3117	816.67	2150	30.17624	991.67	2600	30.05242	1166.67	3050	29.96575
Green	7.075E-02	20	920	544.334	136.67	1520	131.6067	410	2120	61.18706	933.33	2720	34.48588	1133.33	3320	34.66485	1316.67	3920	35.23033
Dark Blue	4.488E-02	42.5	750	329.1952	187.5	1150	114.4136	550	1550	52.57148	1250	1950	29.10086	1500	2350	29.22522	1750	2750	29.31405
RailRoad	1.075E-01	200	800	31.1520															
Utility	5.339E-02	70	300	67.2043															

Same color property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Brown	4.365E-02	3	120	767.2021	15	220	281.3074	45	320	136.3915	135	420	59.67128	240	520	41.55678	350	620	33.97609
Light Blue	6.954E-02	6.67	320	577.5951	33.33	470	169.7704	93.33	620	79.97786	280	770	33.10793	416.67	920	26.58245	566.67	1070	22.7328
Light Purple	7.615E-02	10.67	440	453.3696	53.33	740	152.5542	160	1040	71.46238	466.67	1340	31.56887	650	1640	27.73924	800	1940	26.66097
Orange	8.787E-02	14.67	560	363.7067	73.33	860	111.7404	206.67	1160	53.47785	566.67	1260	21.18526	766.67	1560	19.38695	966.67	1860	18.33277
Red	8.625E-02	18.67	680	353.5408	93.33	1130	117.5255	266.67	1580	57.51192	716.67	2030	27.49488	891.67	2480	26.99743	1066.67	2930	26.66321
Yellow	7.865E-02	22.66	800	375.8075	113.33	1250	117.4087	340	1700	53.22373	816.67	2150	28.02381	991.67	2600	27.90882	1166.67	3050	27.82833
Green	7.727E-02	20	920	498.4034	136.67	1520	120.5018	410	2120	56.02413	933.33	2720	31.57598	1133.33	3320	31.73985	1316.67	3920	32.25762
Dark Blue	4.826E-02	42.5	750	306.1393	187.5	1150	106.4004	550	1550	48.88952	1250	1950	27.06271	1500	2350	27.17837	1750	2750	27.26097
RailRoad	1.138E-01	200	800	29.4274															
Utility	5.405E-02	70	300	66.3837															

Same color property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Brown	4.435E-02	3	120	755.0929	15	220	276.8674	45	320	134.2387	135	420	58.72945	240	520	40.90087	350	620	33.43983
Light Blue	7.152E-02	6.67	320	561.6046	33.33	470	165.0704	93.33	620	77.76371	280	770	32.19135	416.67	920	25.84652	566.67	1070	22.10345
Light Purple	7.765E-02	10.67	440	444.6117	53.33	740	149.6073	160	1040	70.08191	466.67	1340	30.95904	650	1640	27.20339	800	1940	26.14594
Orange	8.763E-02	14.67	560	364.7028	73.33	860	112.0464	206.67	1160	53.62432	566.67	1260	21.24328	766.67	1560	19.44005	966.67	1860	18.38298
Red	8.583E-02	18.67	680	355.2708	93.33	1130	118.1006	266.67	1580	57.79335	716.67	2030	27.62942	891.67	2480	27.12954	1066.67	2930	26.79369
Yellow	7.779E-02	22.66	800	379.9622	113.33	1250	118.7067	340	1700	53.81214	816.67	2150	28.33362	991.67	2600	28.21736	1166.67	3050	28.13598
Green	7.600E-02	20	920	506.7319	136.67	1520	122.5154	410	2120	56.96032	933.33	2720	32.10363	1133.33	3320	32.27024	1316.67	3920	32.79666
Dark Blue	4.853E-02	42.5	750	304.4361	187.5	1150	105.8084	550	1550	48.61752	1250	1950	26.91215	1500	2350	27.02716	1750	2750	27.10931
RailRoad	1.142E-01	200	800	29.3141															
Utility	5.438E-02	70	300	66.1024															

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

Table 11. Ranking the number of turns in Monopoly for properties of the same color with a hotel, using the long jail strategy for both the 43 × 43 model and the 123 × 123 model, and using the short jail strategy for both the 41 × 41 model and the 123 × 123 model.

Rank of Hotel Return	States	Long jail 43×43 model	States	Short jail 41×41 model
1	Brown	$36.10167 \approx 36$	Brown	$33.97609 \approx 34$
2	Green	$34.25238 \approx 34$	Green	$32.25762 \approx 32$
3	Yellow	$29.61302 \approx 30$	Yellow	$27.82833 \approx 28$
4	Dark Blue	$28.88921 \approx 29$	Dark Blue	$27.26097 \approx 27$
5	Red	$28.57838 \approx 29$	Red	$26.66321 \approx 27$
6	Light Purple	$28.09622 \approx 28$	Light Purple	$26.66097 \approx 27$
7	Light Blue	$24.13126 \approx 24$	Light Blue	$22.7328 \approx 23$
8	Orange	$19.44593 \approx 19$	Orange	$18.33277 \approx 18$

Rank of Hotel Return	States	Long jail 123×123 model	States	Short jail 123×123 model
1	Brown	$35.82262 \approx 36$	Brown	$33.43983 \approx 33$
2	Green	$35.23033 \approx 35$	Green	$32.79666 \approx 33$
3	Yellow	$29.96575 \approx 30$	Yellow	$28.13598 \approx 28$
4	Dark Blue	$29.31405 \approx 29$	Dark Blue	$27.10931 \approx 27$
5	Red	$28.15502 \approx 28$	Red	$26.79369 \approx 27$
6	Light Purple	$26.72414 \approx 27$	Light Purple	$26.14594 \approx 26$
7	Light Blue	$23.01746 \approx 23$	Light Blue	$22.10345 \approx 22$
8	Orange	$18.90063 \approx 19$	Orange	$18.38298 \approx 18$

5. Conclusions

In the present study, we have unveiled the mEngel algorithm, which addresses the computation of absorbing and non-absorbing Markov chains through a nested iterative process. We have also furnished proof to demonstrate the efficacy of the mEngel algorithm in resolving non-absorbing Markov chain issues, correlating it with the power method's procedures and the canonical decomposition of absorbing Markov chains. The mEngel algorithm was implemented in a sequential manner using R, Algorithms 1 - 3 being elucidated in detail. Our proposed algorithm has two key features that set it apart from the original Engel algorithm [8,9,21] and the approach in [13]:

1. It can be applied to non-absorbing and absorbing Markov chains, whereas the original Engel algorithm is limited to absorbing cases.
2. It provides process values, such as the passing frequency of intermediate states, which traditional methods cannot provide. This capability allowed us to identify which state had at firing, referred to as a firing state, and record this information for each configuration.

We explored the Engel algorithm's capability to compute the absorbing probabilities for Torrence's problem, with findings that align with established scholarly works. Determining the steady-state probabilities for non-absorbing states in Monopoly, particularly concerning the Jail rules, was also investigated. Using the long jail strategy, the short jail strategy, and the strategy of getting out of Jail by rolling consecutive doubles three times, the 43 × 43 model, the 41 × 41 model, and the 123 × 123 model, respectively, were formulated and tested, and their results are consistent with existing literature. Our findings show that using the short jail strategy is a common practice. Early in the game, being in Jail reduces the player's ability to purchase property, but during the endgame, it protects the player from paying the player's opponent's rent. The player should get out of Jail immediately if the player can pay $50 or use a "Get Out of Jail Free" card at any time from the Community Chest or the Chance card pile, moving the player's token to "Just Visiting". And we gave the rewards of each state and color group under different jail strategies, and players can get their best strategy based on the results. Further research is necessary when considering a case-by-case card evaluation function [27] to assess the value of properties using the mEngel algorithm. The main drawback of the mEngel algorithm is

 doi:10.20944/preprints202507.0451.v1

that it takes longer computational time to achieve high accuracy compared to other methods in the same problem setting. To decrease the computational time, it is essential to reorganize the architecture of Algorithms 1 - 3 by, for example, adjusting the number of iterations, nested loops, and recursive calls. These results will be published elsewhere.

Author Contributions: C.H. L. and J.C.F.W. contributed equally to the research. All authors read and approved the final version of the manuscript.

Funding: This research received no funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

mEngel	modified Engel algorithm
CFG	The Chip-firing Game
TR	Transient
ABS	Absorbing States
M	Movement
J	Jail
Ch	Chance
Cc	Community Chest
L	Long jail
S	Short jail
R	Rent earnings

Appendix A Definitions of transition matrices

This appendix contains all defined transition matrices described in Subsection 4.2.

Table A1. Transition matrix for the long jail strategy.

3/248	0	3/124	12/217	74/793	35/247	30/217	27/434	30/217	24/217	18/217	35/533	23/605	0	0	0	0	0	0	0	0	0	0	0	9/868	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9/868	14/837	0	0
4/463	0	0	6/217	25/391	27/248	24/217	38/733	36/217	30/217	24/217	49/535	25/391	6/217	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4/463	11/829	0	0
3/434	0	0	0	15/434	33/434	18/217	9/217	30/217	36/217	30/217	100/851	70/779	12/217	6/217	0	0	0	0	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/434	11/953	0	0
1/193	0	0	0	1/193	34/787	12/217	27/868	24/217	30/217	36/217	143/997	11/95	18/217	12/217	6/217	0	0	0	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/193	9/917	0	0
3/868	0	0	0	3/868	9/868	6/217	9/434	18/217	24/217	30/217	21/124	35/247	24/217	18/217	12/217	6/217	0	0	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/868	3/371	0	0
3/868	0	0	0	1/579	1/193	0	9/868	12/217	18/217	24/217	111/793	146/871	30/217	24/217	18/217	12/217	3/124	0	0	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/579	3/371	0	0
3/868	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	24/217	18/217	3/62	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/371	0	0
1/193	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	24/217	9/124	12/217	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9/917	0	0
3/434	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	3/31	18/217	12/217	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11/953	0	0
4/463	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	15/124	24/217	18/217	12/217	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11/829	0	0
3/248	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	12/217	18/217	24/217	30/217	9/62	30/217	72/641	18/217	12/217	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	14/837	0	0
3/248	0	0	0	0	3/868	0	0	0	0	0	3/868	0	6/217	12/217	18/217	24/217	15/124	36/217	35/247	24/217	18/217	9/434	6/217	3/868	3/434	0	0	3/868	0	0	0	0	0	0	0	0	0	0	3/868	14/837	0	0
3/248	0	0	0	0	1/193	0	0	0	0	0	1/193	0	0	6/217	12/217	18/217	3/31	30/217	84/491	30/217	24/217	27/868	12/217	11/335	9/868	0	0	1/193	0	0	0	0	0	0	0	0	0	0	1/193	14/837	0	0
3/248	0	0	0	0	3/434	0	0	0	0	0	3/434	0	0	0	6/217	12/217	9/124	24/217	9/62	36/217	30/217	9/217	18/217	27/434	9/217	0	0	3/434	0	0	0	0	0	0	0	0	0	0	3/434	14/837	0	0
3/248	0	0	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	6/217	3/62	18/217	75/629	30/217	36/217	38/733	24/217	49/535	9/124	6/217	0	4/463	0	0	0	0	0	0	0	0	0	0	4/463	14/837	0	0
3/248	0	0	0	0	9/868	0	0	0	0	0	9/868	0	0	0	0	0	3/124	12/217	74/793	24/217	30/217	27/434	30/217	15/124	76/733	12/217	6/217	9/868	0	0	0	0	0	0	0	0	0	0	9/868	14/837	0	0
4/463	0	0	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	0	0	6/217	25/391	18/217	24/217	38/733	36/217	125/851	50/391	18/217	12/217	9/248	0	0	0	0	0	0	0	0	0	0	4/463	11/829	0	0
3/434	0	0	0	0	3/434	0	0	0	0	0	3/434	0	0	0	0	0	0	0	15/434	12/217	18/217	9/217	30/217	136/787	33/217	24/217	18/217	27/434	6/217	0	0	0	0	0	0	0	0	0	3/434	11/953	0	0
1/193	0	0	0	0	1/193	0	0	0	0	0	1/193	0	0	0	0	0	0	0	1/193	6/217	12/217	27/868	24/217	143/997	150/851	30/217	24/217	75/851	12/217	0	0	0	0	0	0	0	0	0	1/193	13/347	0	0
3/868	0	0	0	0	3/868	0	0	0	0	0	3/868	0	0	0	0	0	0	0	3/868	0	6/217	9/434	18/217	56/491	9/62	36/217	30/217	56/491	18/217	0	6/217	0	0	0	0	0	0	0	3/868	9/142	0	0
1/579	0	0	0	0	1/579	0	0	0	0	0	1/579	0	0	0	0	0	0	0	1/579	0	0	9/868	12/217	21/248	56/491	30/217	36/217	111/793	24/217	0	12/217	6/217	0	0	0	0	0	0	1/579	81/907	0	0
1/579	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	0	18/217	12/217	3/124	0	0	0	0	0	0	20/171	0	0
3/868	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	0	24/217	18/217	3/62	6/217	0	0	0	0	0	6/41	0	0
1/193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	0	30/217	24/217	9/124	12/217	6/217	0	0	0	0	68/387	0	0
4/463	0	0	0	0	1/579	0	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	1/579	0	6/217	12/217	21/248	24/217	0	36/217	30/217	46/467	18/217	50/851	9/868	0	0	1/579	5/33	0	0
3/248	0	0	0	0	3/868	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	3/868	0	0	6/217	50/851	18/217	0	30/217	36/217	27/217	24/217	70/779	9/434	6/217	0	3/868	48/377	0	0
10/643	0	0	0	0	1/193	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	1/193	0	0	0	11/335	12/217	0	24/217	30/217	66/439	30/217	15/124	27/868	12/217	6/217	1/193	56/543	0	0
10/643	0	0	0	0	3/434	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	3/434	0	0	0	3/434	6/217	0	18/217	24/217	50/391	36/217	33/217	9/217	18/217	12/217	15/434	4/53	0	0
34/787	0	0	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	4/463	0	0	12/217	18/217	37/351	30/217	135/737	38/733	24/217	18/217	25/391	43/899	0	0
35/494	6/217	0	0	0	9/868	0	0	0	0	0	9/868	0	0	0	0	0	0	0	0	0	0	0	0	9/868	0	0	0	9/868	0	0	6/217	12/217	18/217	24/217	100/629	27/434	30/217	24/217	74/793	20/991	0	0
3/31	12/217	3/124	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	4/463	0	0	0	6/217	28/491	18/217	50/391	38/733	36/217	30/217	75/629	16/867	0	0
113/921	18/217	3/62	6/217	0	3/434	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	3/434	0	0	0	3/434	0	0	0	0	27/868	12/217	3/31	9/217	30/217	36/217	9/62	14/837	0	0
59/397	24/217	9/124	12/217	6/217	1/193	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	1/193	0	0	0	1/193	0	0	0	0	1/193	6/217	35/533	27/868	24/217	30/217	84/491	3/200	0	0
150/851	30/217	3/31	18/217	12/217	27/868	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	3/868	0	0	0	3/868	0	0	0	0	3/868	0	15/434	9/434	18/217	24/217	35/247	3/200	0	0
59/397	36/217	15/124	24/217	18/217	28/491	6/217	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	1/579	0	0	0	1/579	0	0	0	0	1/579	0	3/868	9/868	12/217	18/217	72/641	3/200	0	0
113/921	30/217	9/62	30/217	72/641	75/851	12/217	9/868	0	0	0	1/579	1/579	0	0	0	0	0	0	0	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	21/248	14/837	0	0
23/242	24/217	15/124	36/217	35/247	15/124	18/217	9/434	6/217	0	0	3/868	3/868	0	0	0	0	0	0	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	50/851	14/837	0	0
37/549	18/217	3/31	30/217	84/491	2/13	24/217	27/868	12/217	6/217	0	1/193	1/193	0	0	0	0	0	0	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11/335	14/837	0	0
25/629	12/217	9/124	24/217	9/62	67/359	30/217	9/217	18/217	12/217	6/217	3/434	3/434	0	0	0	0	0	0	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/434	14/837	0	0
3/248	6/217	3/62	18/217	75/629	11/67	36/217	38/733	24/217	18/217	12/217	9/248	4/463	0	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4/463	14/837	0	0
1/579	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	0	6/217	0	6/217	0	6/217	1/579	6/217	0	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	1/400	5/6	0
1/579	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	0	6/217	0	6/217	0	6/217	1/579	6/217	0	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	1/400	0	5/6
3/248	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	12/217	18/217	24/217	30/217	9/62	30/217	72/641	18/217	12/217	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	14/837	0	0

Table A2. Transition matrix for the short jail strategy.

3/248	0	3/124	12/217	74/793	35/247	30/217	27/434	30/217	24/217	18/217	35/533	23/605	0	0	0	0	0	0	0	0	0	0	0	9/868	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9/868	14/837	0	0
4/463	0	0	6/217	25/391	27/248	24/217	38/733	36/217	30/217	24/217	49/535	25/391	6/217	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4/463	11/829	0	0
3/434	0	0	0	15/434	33/434	18/217	9/217	30/217	36/217	30/217	100/851	70/779	12/217	6/217	0	0	0	0	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/434	11/953	0	0
1/193	0	0	0	1/193	34/787	12/217	27/868	24/217	30/217	36/217	143/997	11/95	18/217	12/217	6/217	0	0	0	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/193	9/917	0	0
3/868	0	0	0	3/868	9/868	6/217	9/434	18/217	24/217	30/217	21/124	35/247	24/217	18/217	12/217	6/217	0	0	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/868	3/371	0	0
3/868	0	0	0	1/579	1/193	0	9/868	12/217	18/217	24/217	111/793	146/871	30/217	24/217	18/217	12/217	3/124	0	0	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/579	3/371	0	0
3/868	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	24/217	18/217	3/62	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/371	0	0
1/193	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	24/217	9/124	12/217	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9/917	0	0
3/434	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	3/31	18/217	12/217	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11/953	0	0
4/463	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	15/124	24/217	18/217	12/217	6/217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11/829	0	0
3/248	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	12/217	18/217	24/217	30/217	9/62	30/217	72/641	18/217	12/217	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	14/837	0	0
3/248	0	0	0	0	3/868	0	0	0	0	0	3/868	0	6/217	12/217	18/217	24/217	15/124	36/217	35/247	24/217	18/217	9/434	6/217	3/868	3/434	0	0	3/868	0	0	0	0	0	0	0	0	0	0	3/868	14/837	0	0
3/248	0	0	0	0	1/193	0	0	0	0	0	1/193	0	0	6/217	12/217	18/217	3/31	30/217	84/491	30/217	24/217	27/868	12/217	11/335	9/868	0	0	1/193	0	0	0	0	0	0	0	0	0	0	1/193	14/837	0	0
3/248	0	0	0	0	3/434	0	0	0	0	0	3/434	0	0	0	6/217	12/217	9/124	24/217	9/62	36/217	30/217	9/217	18/217	27/434	9/217	0	0	3/434	0	0	0	0	0	0	0	0	0	0	3/434	14/837	0	0
3/248	0	0	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	6/217	3/62	18/217	75/629	30/217	36/217	38/733	24/217	49/535	9/124	6/217	0	4/463	0	0	0	0	0	0	0	0	0	0	4/463	14/837	0	0
3/248	0	0	0	0	9/868	0	0	0	0	0	9/868	0	0	0	0	0	3/124	12/217	74/793	24/217	30/217	27/434	30/217	15/124	76/733	12/217	6/217	9/868	0	0	0	0	0	0	0	0	0	0	9/868	14/837	0	0
4/463	0	0	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	0	0	6/217	25/391	18/217	24/217	38/733	36/217	125/851	50/391	18/217	12/217	9/248	0	0	0	0	0	0	0	0	0	0	4/463	11/829	0	0
3/434	0	0	0	0	3/434	0	0	0	0	0	3/434	0	0	0	0	0	0	0	15/434	12/217	18/217	9/217	30/217	136/787	33/217	24/217	18/217	27/434	6/217	0	0	0	0	0	0	0	0	0	3/434	11/953	0	0
1/193	0	0	0	0	1/193	0	0	0	0	0	1/193	0	0	0	0	0	0	0	1/193	6/217	12/217	27/868	24/217	143/997	150/851	30/217	24/217	75/851	12/217	0	0	0	0	0	0	0	0	0	1/193	13/347	0	0
3/868	0	0	0	0	3/868	0	0	0	0	0	3/868	0	0	0	0	0	0	0	3/868	0	6/217	9/434	18/217	56/491	9/62	36/217	30/217	56/491	18/217	0	6/217	0	0	0	0	0	0	0	3/868	9/142	0	0
1/579	0	0	0	0	1/579	0	0	0	0	0	1/579	0	0	0	0	0	0	0	1/579	0	0	9/868	12/217	21/248	56/491	30/217	36/217	111/793	24/217	0	12/217	6/217	0	0	0	0	0	0	1/579	81/907	0	0
1/579	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	30/217	0	18/217	12/217	3/124	0	0	0	0	0	0	20/171	0	0
3/868	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	36/217	0	24/217	18/217	3/62	6/217	0	0	0	0	0	6/41	0	0
1/193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	18/217	24/217	30/217	0	30/217	24/217	9/124	12/217	6/217	0	0	0	0	68/387	0	0
4/463	0	0	0	0	1/579	0	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	1/579	0	6/217	12/217	21/248	24/217	0	36/217	30/217	46/467	18/217	50/851	9/868	0	0	1/579	5/33	0	0
3/248	0	0	0	0	3/868	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	3/868	0	0	6/217	50/851	18/217	0	30/217	36/217	27/217	24/217	70/779	9/434	6/217	0	3/868	48/377	0	0
10/643	0	0	0	0	1/193	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	1/193	0	0	0	11/335	12/217	0	24/217	30/217	66/439	30/217	15/124	27/868	12/217	6/217	1/193	56/543	0	0
10/643	0	0	0	0	3/434	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	3/434	0	0	0	3/434	6/217	0	18/217	24/217	50/391	36/217	33/217	9/217	18/217	12/217	15/434	4/53	0	0
34/787	0	0	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	4/463	0	0	12/217	18/217	37/351	30/217	135/737	38/733	24/217	18/217	25/391	43/899	0	0
35/494	6/217	0	0	0	9/868	0	0	0	0	0	9/868	0	0	0	0	0	0	0	0	0	0	0	0	9/868	0	0	0	9/868	0	0	6/217	12/217	18/217	24/217	100/629	27/434	30/217	24/217	74/793	20/991	0	0
3/31	12/217	3/124	0	0	4/463	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	4/463	0	0	0	6/217	28/491	18/217	50/391	38/733	36/217	30/217	75/629	16/867	0	0
113/921	18/217	3/62	6/217	0	3/434	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	3/434	0	0	0	3/434	0	0	0	0	27/868	12/217	3/31	9/217	30/217	36/217	9/62	14/837	0	0
59/397	24/217	9/124	12/217	6/217	1/193	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	1/193	0	0	0	1/193	0	0	0	0	1/193	6/217	35/533	27/868	24/217	30/217	84/491	3/200	0	0
150/851	30/217	3/31	18/217	12/217	27/868	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	3/868	0	0	0	3/868	0	0	0	0	3/868	0	15/434	9/434	18/217	24/217	35/247	3/200	0	0
59/397	36/217	15/124	24/217	18/217	28/491	6/217	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	1/579	0	0	0	1/579	0	0	0	0	1/579	0	3/868	9/868	12/217	18/217	72/641	3/200	0	0
113/921	30/217	9/62	30/217	72/641	75/851	12/217	9/868	0	0	0	1/579	1/579	0	0	0	0	0	0	0	0	0	0	0	1/579	0	0	0	0	0	0	0	0	0	0	0	0	6/217	12/217	21/248	14/837	0	0
23/242	24/217	15/124	36/217	35/247	15/124	18/217	9/434	6/217	0	0	3/868	3/868	0	0	0	0	0	0	0	0	0	0	0	3/868	0	0	0	0	0	0	0	0	0	0	0	0	0	6/217	50/851	14/837	0	0
37/549	18/217	3/31	30/217	84/491	2/13	24/217	27/868	12/217	6/217	0	1/193	1/193	0	0	0	0	0	0	0	0	0	0	0	1/193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11/335	14/837	0	0
25/629	12/217	9/124	24/217	9/62	67/359	30/217	9/217	18/217	12/217	6/217	3/434	3/434	0	0	0	0	0	0	0	0	0	0	0	3/434	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3/434	14/837	0	0
3/248	6/217	3/62	18/217	75/629	11/67	36/217	38/733	24/217	18/217	12/217	9/248	4/463	0	0	0	0	0	0	0	0	0	0	0	4/463	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4/463	14/837	0	0
3/248	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	12/217	18/217	24/217	30/217	9/62	30/217	72/641	18/217	12/217	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	14/837	0	0
3/248	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	12/217	18/217	24/217	30/217	9/62	30/217	72/641	18/217	12/217	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	14/837	0	0
3/248	0	0	0	0	1/579	0	0	0	0	0	1/579	6/217	12/217	18/217	24/217	30/217	9/62	30/217	72/641	18/217	12/217	9/868	0	1/579	3/868	0	0	1/579	0	0	0	0	0	0	0	0	0	0	1/579	14/837	0	0

Table A3. Movement matrix for the long jail strategy.

0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0
1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	0	0	0
1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	0	0	0
1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	0	0	0
1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	0	0	0
5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	0	0	0
1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	0	0	0
5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	0	0	0
1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	0	0	0
1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	0	0	0
1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	0	0	0
1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5/6	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36	0	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5/6
0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A4. Movement matrix for the short jail strategy.

0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0
1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	0	0	0
1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	0	0	0
1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	0	0	0
1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	0	0	0
5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	0	0	0
1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	0	0	0
5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	0	0	0
1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	0	0	0
1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	0	0	0
1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/36	0	0	0
1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1/36	1/18	1/12	1/9	5/36	1/6	5/36	1/9	1/12	1/18	1/36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A5. Jail matrix.

215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	0	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	215/216	1/216	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

Table A6. Chance matrix.

1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1/16	0	0	0	1/16	3/16	0	3/8	0	0	0	1/16	1/16	0	0	0	0	0	0	0	0	0	0	0	1/16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/16	1/16	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1/16	0	0	0	0	1/16	0	0	0	0	0	1/16	0	0	0	0	0	0	0	1/16	0	0	3/8	0	1/16	1/8	0	0	1/16	0	0	0	0	0	0	0	0	0	0	1/16	1/16	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
1/16	0	0	0	0	1/16	0	0	0	0	0	1/16	0	0	0	0	0	0	0	0	0	0	0	0	1/16	0	0	0	1/16	0	0	0	0	1/16	0	1/8	3/8	0	0	1/16	1/16	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Table A7. Community Chest matrix.

1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1/16	0	7/8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/16	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1/16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7/8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1/16	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
1/16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7/8	0	0	0	0	0	0	1/16	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Appendix B Turns in Monopoly for each property using different strategies

Please refer to Subsection 4.4 for a calculation of the number of turns required to generate revenue for each property in Monopoly. This appendix includes data for both the 43×43 model and the 123×123 model, using the long jail strategy, as well as data for the 41×41 model and the 123×123 model, using the short jail strategy.

 doi:10.20944/preprints202507.0451.v1

Table A8. Turns in Monopoly for each property using the long jail strategy for the 43 × 43 model.

Property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Go	2.9220E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Mediterranean Avenue	2.0355E-02	2	60	1233.914	10	10	699.2179	30	220	301.6234	90	270	123.3914	160	320	82.26093	250	370	60.87309
Community Chest (South)	1.8052E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Baltic Avenue	2.0722E-02	4	60	606.0323	20	20	343.4183	60	220	148.1412	180	270	60.60323	320	320	40.40215	450	370	33.21955
Income Tax	2.2233E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Reading RailRoad	2.8312E-02	25	200	236.5666	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Oriental Avenue	2.1544E-02	6	100	647.6727	30	30	479.2778	90	420	181.3484	270	470	67.64581	400	520	50.51847	550	570	40.27347
Chance (South)	8.2430E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Vermont Avenue	2.2046E-02	6	100	632.9215	30	30	468.3619	90	420	177.218	270	470	66.10514	400	520	49.36788	550	570	39.35621
Connecticut Avenue	2.1919E-02	8	120	572.9275	40	40	353.3053	100	420	160.4197	300	470	59.83909	450	520	44.13663	600	570	36.28541
Just Visiting	2.1630E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Jail (First turn)	3.7755E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Jail (Second turn)	3.1430E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Jail (Third turn)	2.6192E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. Charles Place	2.5807E-02	10	140	454.1722	50	50	350.3614	150	640	138.4144	450	740	53.34722	625	840	43.60054	750	940	40.65923
Electric Company	2.5050E-02	28	150	179.0439	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
States Avenue	2.1999E-02	10	140	532.7875	50	50	411.0075	150	640	162.3733	450	740	62.58139	625	840	51.1476	750	940	47.69717
Virginia Avenue	2.4455E-02	12	160	456.469	60	60	308.1166	180	640	121.7251	500	740	50.66806	700	840	41.08221	900	940	35.75674
Pennsylvania RailRoad	2.3798E-02	25	200	281.4385	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. James Place	2.6921E-02	14	180	399.8397	70	70	293.2158	200	760	118.1748	550	860	48.62697	750	960	39.80626	950	1060	34.69954
Community Chest (West)	2.2957E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Tennessee Avenue	2.8094E-02	14	180	383.1519	70	70	280.9781	200	760	113.2427	550	860	46.59746	750	960	38.1449	950	1060	33.25131
New York Avenue	2.7827E-02	16	200	376.0833	80	80	248.215	220	760	103.9357	600	860	43.12421	800	960	36.10399	1000	1060	31.89186
Free Parking	2.7947E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Kentucky Avenue	2.5740E-02	18	220	397.5308	90	90	299.9551	250	980	127.499	700	1130	52.50504	875	1280	47.57979	1050	1430	44.29629
Chance (North)	1.0270E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Indiana Avenue	2.5216E-02	18	220	405.7946	90	90	306.1905	250	980	130.1494	700	1130	53.59651	875	1280	48.56887	1100	1430	43.16179
Illinois Avenue	2.9513E-02	20	240	340.414	100	100	235.453	300	980	92.66825	750	1130	42.74086	925	1280	39.25494	1150	1430	35.27478
B & O RailRoad	2.8496E-02	25	200	235.0391	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Atlantic Avenue	2.5031E-02	22	260	395.2774	110	110	288.8566	330	1100	111.4885	800	1250	52.26024	975	1400	48.02582	1200	1550	43.2018
Ventnor Avenue	2.4844E-02	22	260	398.2623	110	110	291.0379	330	1100	112.3304	800	1250	52.65487	975	1400	48.38848	1275	1550	40.96756
Water Works	2.7545E-02	28	150	162.8263	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Marvin Gardens	2.4035E-02	24	280	406.3789	120	120	275.7571	360	1100	106.4326	850	1250	51.22424	1025	1400	47.57607	1275	1550	42.34537
Pacific Avenue	2.4926E-02	26	300	387.5449	130	130	289.3669	390	1320	113.6798	900	1520	56.72509	1100	1720	52.51821	1400	1920	46.06248
North Carolina Avenue	2.4466E-02	26	300	394.843	130	130	294.8161	390	1320	115.8206	900	1520	57.79332	1100	1720	53.50721	1100	1920	59.72898
Community Chest (East)	2.2259E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Pennsylvania Avenue	2.3381E-02	28	320	409.2178	150	150	267.3556	450	1320	105.0326	1000	1520	54.42597	1200	1720	51.32273	1400	1920	49.10614
Short Line RailRoad	2.5533E-02	25	200	262.3144	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Chance (East)	8.1280E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Park Place	2.0598E-02	35	350	406.4567	175	175	220.6479	500	1150	93.48504	1100	1350	49.88332	1300	1550	48.46214	1500	1750	47.41995
Luxury Tax	2.0565E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Boardwalk	2.4946E-02	50	400	268.4822	200	200	159.4113	600	1150	64.32386	1400	1350	32.3617	1700	1550	30.59908	2000	1750	29.36524

doi:10.20944/preprints202507.0451.v1

Table A9. Turns in Monopoly for each property using the long jail strategy for the 123 × 123 model.

Property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Go	2.9160E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Mediterranean Avenue	2.0362E-02	2	60	1233.509	10	170	698.9886	30	220	301.5245	90	270	123.3509	160	320	82.23395	250	370	60.85312
Community Chest (South)	1.8191E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Baltic Avenue	2.1038E-02	4	60	596.9153	20	170	338.252	60	220	145.9126	180	270	59.69153	320	320	39.79435	450	370	32.7198
Income Tax	2.2788E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Reading RailRoad	2.9144E-02	25	200	229.8131	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Oriental Avenue	2.2373E-02	6	100	623.6814	30	370	461.5242	90	420	174.6308	270	470	65.14006	400	520	48.64715	550	570	38.78164
Chance (South)	8.6140E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Vermont Avenue	2.3168E-02	6	100	602.2746	30	370	445.6832	90	420	168.6369	270	470	62.90424	400	520	46.97742	550	570	37.45053
Connecticut Avenue	2.3137E-02	8	120	542.7636	40	370	334.7042	100	420	151.9738	300	470	56.68865	450	520	41.8129	600	570	34.37503
Just Visiting	2.2933E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Jail	8.3424E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. Charles Place	2.7218E-02	10	140	430.6359	50	540	332.2049	150	640	131.2414	450	740	50.58263	625	840	41.34105	750	940	38.55217
Electric Company	2.6236E-02	28	150	170.9502	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
States Avenue	2.3298E-02	10	140	503.0897	50	540	388.0978	150	640	153.3226	450	740	59.09308	625	840	48.29661	750	940	45.03851
Virginia Avenue	2.5452E-02	12	160	438.5784	60	540	296.0404	180	640	116.9542	500	740	48.6822	700	840	39.47205	900	940	34.35531
Pennsylvania RailRoad	2.4969E-02	25	200	268.2396	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. James Place	2.7770E-02	14	180	387.6203	70	660	284.2549	200	760	114.5633	550	860	47.14089	750	960	38.58975	950	1060	33.63909
Community Chest (West)	2.3791E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Tennessee Avenue	2.8775E-02	14	180	374.0779	70	660	274.3238	200	760	110.5608	550	860	45.49392	750	960	37.24154	950	1060	32.46384
New York Avenue	2.8685E-02	16	200	364.8294	80	660	240.7874	220	760	100.8256	600	860	41.83377	800	960	35.02362	1000	1060	30.93753
Free Parking	2.8474E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Kentucky Avenue	2.6364E-02	18	220	388.1295	90	830	292.8613	250	980	124.4837	700	1130	51.26333	875	1280	46.45456	1050	1430	43.24871
Chance (North)	1.0369E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Indiana Avenue	2.5571E-02	18	220	400.163	90	830	301.9412	250	980	128.3432	700	1130	52.8527	875	1280	47.89483	1100	1430	42.56279
Illinois Avenue	2.9743E-02	20	240	337.7788	100	830	233.6303	300	980	91.9509	750	1130	42.41001	925	1280	38.95107	1150	1430	35.00172
B & O RailRoad	2.8552E-02	25	200	234.5781	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Atlantic Avenue	2.4905E-02	22	260	397.2735	110	950	290.3153	330	1100	112.0515	800	1250	52.52414	975	1400	48.26834	1200	1550	43.41996
Ventnor Avenue	2.4582E-02	22	260	402.4978	110	950	294.133	330	1100	113.525	800	1250	53.21485	975	1400	48.90308	1275	1550	41.40324
Water Works	2.7150E-02	28	150	165.1952	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Marvin Gardens	2.3557E-02	24	280	414.625	120	950	281.3527	360	1100	108.5923	850	1250	52.26366	1025	1400	48.54146	1275	1550	43.20462
Pacific Avenue	2.4270E-02	26	300	398.0252	130	1120	297.1922	390	1320	116.7541	900	1520	58.2591	1100	1720	53.93845	1400	1920	47.30814
North Carolina Avenue	2.3776E-02	26	300	406.3009	130	1120	303.3713	390	1320	119.1816	900	1520	59.47041	1100	1720	55.05992	1100	1920	61.46224
Community Chest (East)	2.1616E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Pennsylvania Avenue	2.2704E-02	28	320	421.425	150	1120	275.331	450	1320	108.1657	1000	1520	56.04952	1200	1720	52.85372	1400	1920	50.571
Short Line RailRoad	2.4834E-02	25	200	269.6978	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Chance (East)	7.9220E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Park Place	2.0149E-02	35	350	415.5094	175	950	225.5623	500	1150	95.56717	1100	1350	50.99434	1300	1550	49.54151	1500	1750	48.4761
Luxury Tax	2.0201E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Boardwalk	2.4734E-02	50	400	270.7882	200	950	160.7805	600	1150	64.87635	1400	1350	32.63965	1700	1550	30.86189	2000	1750	29.61746

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

Table A10. Turns in Monopoly for each property using the short jail strategy for the 41 × 41 model.

Property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Go	3.1050E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Mediterranean Avenue	2.1629E-02	2	60	1161.215	10	170	658.0216	30	220	283.8525	90	270	116.1215	160	320	77.41431	250	370	57.28659
Community Chest (South)	1.9181E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Baltic Avenue	2.2016E-02	4	60	570.4161	20	170	323.2358	60	220	139.435	180	270	57.04161	320	320	38.02774	450	370	31.26725
Income Tax	2.3617E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Reading RailRoad	2.9964E-02	25	200	223.5240	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Oriental Avenue	2.2874E-02	6	100	610.0102	30	370	451.4075	90	420	170.8028	270	470	63.71217	400	520	47.58079	550	570	37.93154
Chance (South)	8.7520E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Vermont Avenue	2.3404E-02	6	100	596.197	30	370	441.1858	90	420	166.9352	270	470	62.26946	400	520	46.50336	550	570	37.07261
Connecticut Avenue	2.3266E-02	8	120	539.7602	40	370	332.8521	100	420	151.1329	300	470	56.37495	450	520	41.58153	600	570	34.18481
Just Visiting	2.2954E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Jail	4.0069E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. Charles Place	2.7281E-02	10	140	429.6315	50	540	331.43	150	640	130.9353	450	740	50.46465	625	840	41.24463	750	940	38.46225
Electric Company	2.4882E-02	28	150	180.2528	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
States Avenue	2.4003E-02	10	140	488.3084	50	540	376.695	150	640	148.8178	450	740	57.35686	625	840	46.8776	750	940	43.71523
Virginia Avenue	2.4864E-02	12	160	448.948	60	540	303.0399	180	640	119.7195	500	740	49.83323	700	840	40.40532	900	940	35.16759
Pennsylvania RailRoad	2.6501E-02	25	200	252.7329	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. James Place	2.8063E-02	14	180	383.5635	70	660	281.2799	200	760	113.3643	550	860	46.64752	750	960	38.18588	950	1060	33.28703
Community Chest (West)	2.5970E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Tennessee Avenue	2.9245E-02	14	180	368.0689	70	660	269.9172	200	760	108.7848	550	860	44.76313	750	960	36.64331	950	1060	31.94236
New York Avenue	3.0561E-02	16	200	342.4284	80	660	226.0027	220	760	94.63476	600	860	39.26512	800	960	32.87313	1000	1060	29.03793
Free Parking	2.8516E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Kentucky Avenue	2.7948E-02	18	220	366.1346	90	830	276.2652	250	980	117.4294	700	1130	48.3583	875	1280	43.82204	1050	1430	40.79786
Chance (North)	1.0290E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Indiana Avenue	2.6885E-02	18	220	380.6071	90	830	287.1854	250	980	122.0711	700	1130	50.26979	875	1280	45.55422	1100	1430	40.48275
Illinois Avenue	3.1416E-02	20	240	319.7852	100	830	221.1847	300	980	87.05263	750	1130	40.1508	925	1280	36.87613	1150	1430	33.13716
B & O RailRoad	3.0215E-02	25	200	221.6672	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Atlantic Avenue	2.6705E-02	22	260	370.4976	110	950	270.7482	330	1100	104.4993	800	1250	48.98405	975	1400	45.01509	1200	1550	40.49348
Ventnor Avenue	2.6435E-02	22	260	374.2888	110	950	273.5187	330	1100	105.5686	800	1250	49.4853	975	1400	45.47572	1275	1550	38.5015
Water Works	2.9165E-02	28	150	153.7819	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Marvin Gardens	2.5507E-02	24	280	382.9248	120	950	259.8418	360	1100	100.2898	850	1250	48.26783	1025	1400	44.83022	1275	1550	39.90141
Pacific Avenue	2.6460E-02	26	300	365.0868	130	1120	272.5981	390	1320	107.0921	900	1520	53.43789	1100	1720	49.47479	1400	1920	43.39317
North Carolina Avenue	2.5974E-02	26	300	371.9152	130	1120	277.6967	390	1320	109.0951	900	1520	54.43737	1100	1720	50.40015	1100	1920	56.26063
Community Chest (East)	2.3655E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Pennsylvania Avenue	2.4837E-02	28	320	385.2402	150	1120	251.6903	450	1320	98.87832	1000	1520	51.23695	1200	1720	48.31554	1400	1920	46.22883
Short Line RailRoad	2.7122E-02	25	200	246.9462	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Chance (East)	8.6310E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Park Place	2.1869E-02	35	350	382.8212	175	950	207.8172	500	1150	88.04888	1100	1350	46.9826	1300	1550	45.64407	1500	1750	44.66248
Luxury Tax	2.1834E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Boardwalk	2.6387E-02	50	400	253.8248	200	950	150.7085	600	1150	60.81219	1400	1350	30.59495	1700	1550	28.92856	2000	1750	27.76209

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 July 2025 doi:10.20944/preprints202507.0451.v1

Table A11. Turns in Monopoly for each property using the short jail strategy for the 123 × 123 model.

Property	Probability	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn	Rent	Cost	Turn
Go	3.1326E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Mediterranean Avenue	2.1925E-02	2	60	1145.554	10	170	649.147	30	220	280.0242	90	270	114.5554	160	320	76.37024	250	370	56.51398
Community Chest (South)	1.9492E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Baltic Avenue	2.2425E-02	4	60	560.0061	20	170	317.3368	60	220	136.8904	180	270	56.00061	320	320	37.33374	450	370	30.69663
Income Tax	2.4130E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Reading RailRoad	3.0606E-02	25	200	218.8353	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Oriental Avenue	2.3464E-02	6	100	594.6796	30	370	440.0629	90	420	166.5103	270	470	62.11099	400	520	46.38501	550	570	36.97826
Chance (South)	8.9930E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Vermont Avenue	2.4088E-02	6	100	579.2758	30	370	428.6641	90	420	162.1972	270	470	60.50214	400	520	45.18351	550	570	36.02042
Connecticut Avenue	2.3972E-02	8	120	523.8757	40	370	323.0567	100	420	146.6852	300	470	54.7159	450	520	40.35783	600	570	33.17879
Just Visiting	2.3671E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Jail	3.5222E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. Charles Place	2.8034E-02	10	140	418.0931	50	540	322.529	150	640	127.4189	450	740	49.10935	625	840	40.13694	750	940	37.42929
Electric Company	2.5514E-02	28	150	175.7878	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
States Avenue	2.4455E-02	10	140	479.2794	50	540	369.7298	150	640	146.0661	450	740	56.29631	625	840	46.01082	750	940	42.90692
Virginia Avenue	2.5165E-02	12	160	443.5865	60	540	299.4209	180	640	118.2897	500	740	49.23811	700	840	39.92279	900	940	34.74761
Pennsylvania RailRoad	2.6662E-02	25	200	251.2068	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
St. James Place	2.8065E-02	14	180	383.5388	70	660	281.2618	200	760	113.357	550	860	46.64451	750	960	38.18341	950	1060	33.28488
Community Chest (West)	2.5814E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Tennessee Avenue	2.9115E-02	14	180	369.7133	70	660	271.1231	200	760	109.2708	550	860	44.96311	750	960	36.80701	950	1060	32.08506
New York Avenue	3.0448E-02	16	200	343.7042	80	660	226.8448	220	760	94.98736	600	860	39.41142	800	960	32.99561	1000	1060	29.14612
Free Parking	2.8416E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Kentucky Avenue	2.7836E-02	18	220	367.6007	90	830	277.3715	250	980	117.8996	700	1130	48.55194	875	1280	43.99751	1050	1430	40.96122
Chance (North)	1.0243E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Indiana Avenue	2.6761E-02	18	220	382.3616	90	830	288.5092	250	980	122.6338	700	1130	50.50153	875	1280	45.76422	1100	1430	40.66937
Illinois Avenue	3.1233E-02	20	240	321.6596	100	830	222.4812	300	980	87.56289	750	1130	40.38615	925	1280	37.09228	1150	1430	33.33139
B & O RailRoad	2.9929E-02	25	200	223.7854	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Atlantic Avenue	2.6411E-02	22	260	374.6289	110	950	273.7673	330	1100	105.6646	800	1250	49.53026	975	1400	45.51704	1200	1550	40.94502
Ventnor Avenue	2.6132E-02	22	260	378.6304	110	950	276.6915	330	1100	106.7932	800	1250	50.05931	975	1400	46.00322	1275	1550	38.94811
Water Works	2.8870E-02	28	150	155.3534	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Marvin Gardens	2.5245E-02	24	280	386.9068	120	950	262.5439	360	1100	101.3327	850	1250	48.76976	1025	1400	45.2964	1275	1550	40.31633
Pacific Avenue	2.6226E-02	26	300	368.3453	130	1120	275.0312	390	1320	108.048	900	1520	53.91484	1100	1720	49.91637	1400	1920	43.78047
North Carolina Avenue	2.5762E-02	26	300	374.9787	130	1120	279.9841	390	1320	109.9938	900	1520	54.88577	1100	1720	50.8153	1100	1920	56.72405
Community Chest (East)	2.3498E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Pennsylvania Avenue	2.4701E-02	28	320	387.3535	150	1120	253.071	450	1320	99.42074	1000	1520	51.51802	1200	1720	48.58059	1400	1920	46.48242
Short Line RailRoad	2.7040E-02	25	200	247.6951	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Chance (East)	8.6250E-03	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Park Place	2.1925E-02	35	350	381.8509	175	950	207.2905	500	1150	87.8257	1100	1350	46.86352	1300	1550	45.52837	1500	1750	44.54927
Luxury Tax	2.1955E-02	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA	0	0	NA
Boardwalk	2.6606E-02	50	400	251.7395	200	950	149.4703	600	1150	60.31259	1400	1350	30.3436	1700	1550	28.6909	2000	1750	27.53401

References

1. Abbott, S. D., Richey, M. Take a walk on the boardwalk. The College Mathematics Journal, **28**(3): 162-171, 1997.
2. Ash, R., Bishop, R. Monopoly as a Markov process. The College Mathematics Journal, **45**(1): 26-29, 1972.
3. Bilisoly, R. Using Board Games and Mathematica to Teach the Fundamentals of Finite Stationary Markov Chains. Section on Statistical Education, 2014. Available at: https://arxiv.org/abs/1410.1107
4. Betz, K. Go Directly To Jail: The Mathematics Behind Family Game Night, BSc. Thesis, Albright College, Department of Mathematics, 2019. Available at: https://libservices.albright.edu/digital-archive/collection.pdfreader/35?fid=9669.
5. Bewersdorff, J. Luck, logic, and white lies: The mathematics of games. Boca Raton, FL: CRC Press, 2004.
6. Collins, T. Probabilities in the Game of Monopoly, 1997. Available at: http://www.tkcs-collins.com/truman/monopoly/monopoly.shtml
7. Eithun, M. Lamb, M. Young, A. Long-term vs short-term strategy in the game of Monopoly UMAP J., **38**(1): 35-57, 2017.
8. Engel, A. The probabilistic abacus. Educational Studies in Mathematics, **6**(1): 1–22, 1975. doi.org/10.1007/bf00590021
9. Engel. A. Why does the probabilistic abacus work? Educational Studies in Mathematics, **7**: 59–69, 1976. doi.org/10.1007/BF00144359
10. Friedman, E., Henderson, S., Byuen, T., & Gallardo, G. Estimating the probability that the game of Monopoly never ends. Paper presented at the Proceedings of the 2009 Winter Simulation Conference, Cornell University, New York: Schools of Operations Research and Information Engineering, 2009.
11. Johnson, R. W. Using games to teach Markov chains. PRIMUS: Problems, Resources and Issues in Mathematics Undergraduate Studies XIII, **13**(4):, 337-348, 2003. DOI: 10.1080/10511970308984067
12. Murrell, P. R. The Statistics of Monopoly, CHANCE, **12**(4), 36-40, 1999. DOI: 10.1080/09332480.1999.10542173
13. Kaushal, N., Tiwari, M., Singh, V., Parihar, C. L. Generalized Engel's Algorithm for Minimizing Playing Time to Stabilize the Initial Configuration and for finding Absorbing Probability, International Journal of Computer Applications, **107**(5): 32-35, 2014.
14. Kaushal, N., Tiwari, M., Parihar, C. L. Chip-Firing Game as Probability Abacus by Characterization of ULD Lattice. Asian Journal of Mathematics and Applications, 8 pages, 2014. Article ID ama0167.
15. Kaushal, N., Tiwari, M., Singh, V., Parihar, C. L. A study of distributive lattices and related computer programming, 2014. Available at: http://hdl.handle.net/10603/239119
16. Li, B. Markov Chains in the Game of Monopoly, Lecture note, unpublished, 2013.
17. Nilsson, A. Exploring strategies in Monopoly using Markov chains and simulation, 2020. U.U.D.M. Project Report 2020:42
18. Osborne, J. A. Markov Chains for the RISK Board Game Revisited. Mathematics Magazine, **76**(2): 129-135, 2003. DOI: 10.1080/0025570X.2003.11953165
19. Shrestha, S. R., Lewin, R. A., Seitzer, J.Analyzing Real Estate Implications of Monopoly, Mega Edition through Simulation. AEF Papers & Proceedings, **39**: 37-39, 2015.
20. Snell, J. L. Finite Markov Chains and their Applications. The American Mathematical Monthly, **66**(2): 99-104, 1959.
21. Snell, J. L. The Engel algorithm for absorbing Markov chains, 1979. Available at: https://arxiv.org/abs/0904.1413v1
22. Propp, J. Engel's Marvelously Improbable Machines. Math Horizons, **26**(2): 5-9, 2018. DOI: https://doi.org/10.1080/10724117.2018.1518840
23. Tan, B. Markov Chains and the RISK Board Game. Mathematics Magazine, **70**(5): 349-357, 1997. DOI: 10.1080/0025570X.1997.11996573
24. Taylor, D. G. Games, Gambling, and Probability - An Introduction to Mathematics. Boca Raton, FL: CRC Press, 2021.
25. Roeder, O. Who Keeps The Money You Found On The Floor?, accessed 9 July 2016, https://fivethirtyeight.com/features/who-keeps-the-money-you-found-on-the-floor/
26. Wu, D. W., Baeth, N. How often does a monopoly player go to 'JAIL'? International Journal of Mathematical Education in Science and Technology, **32**(5): 774-778, 2010. DOI: 10.1080/002073901753124646
27. Yasumura, Y., Oguchi, K., Nitta, K. Negotiation strategy of agents in the Monopoly game. In: Proceedings 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation, pp. 277–281, 2001.

doi:10.20944/preprints202507.0451.v1