2011

A Simultaneous Random Walk Game

Jorge Bañuelos
Macalester College

Honors Project

Macalester College

Spring 2011

Title: A Simultaneous Random Walk Game

Author: Jorge Bañuelos

A Simultaneous Random Walk Game

Jorge H. Bañuelos

Andrew Beveridge, Advisor

Alicia Johnson, Reader

Mike Weimerskirch, Reader

May, 2011

Macalester College

Department of Mathematics, Statistics and Computer Science

Abstract

Two tokens are placed on vertices of a graph. At each time step, one token is chosen and is moved to a random neighboring point. In previous work, Tetali and Winkler studied the Angel strategy for bringing the tokens together as quickly as possible (on average), and the Demon strategy for delaying their collision as long as possible (on average). We build on these results by studying a game version of this process.

In our game, two players take turns choosing the token to move. The Angel player hopes to bring the tokens together while the Demon player tries to keep them apart. We present optimal strategies for both players on stars, different types of directed cycles, and paths. Our proofs employ couplings of random walks as well as strategy stealing arguments.

Contents

Chapter 1

Introduction

In this paper, we study games that require the players to make optimal choices under uncertainty. More precisely, we study the evolution of processes in which each round consists of a strategic choice by a player, followed by a random move. We explore how the decisions of the player influences the evolution of the process. In particular, each player will have a goal, and we want to characterize the optimal strategy for achieving this goal.

Our game is directly inspired by the work of Coppersmith, Tetali and Winkler in Coppersmith et al. (1993b) and Tetali and Winkler (1993). Two tokens are placed on a graph. At each time step, one of the tokens is chosen and it moves to a randomly chosen neighboring point. We refer to the entity that chooses the token that will move as a *player*. The sequence of decisions made by the player is a *strategy*. In Coppersmith et al. (1993b), and Tetali and Winkler (1993), the authors study the Angel strategy for bringing the tokens together as quickly as possible, and the Demon strategy for delaying their collision as long as possible. In this paper we expand on those two papers by considering the game version of this process. In this game, Angel and Demon alternate turns picking the token that will move next. Both players still maintain their original goals. The Angel player tries to bring the tokens together while the Demon player tries to keep them apart.

In this paper, we completely determine the optimal game strategies for Angel and Demon on stars, paths, and certain directed cycles. We primarily use three techniques: coupling, strategy stealing and the introduction of additional players (using non-optimal strategies). Coupling aligns two random processes, so they may evolve together. This allows us to methodically compare these two processes. Strategy stealing provides us with a tool to compare two optimal strategies in different game instances. Sometimes we compare strategies starting from identical game configurations, other times we start in distinct (but similar) game configurations. With strategy stealing, we can show that one strategy superior to another. Our last technique is to introduce players that use hybrid, non-optimal strategies. We use these hybrid strategies to prove that a given concrete strategy is in fact optimal. We usually refer to a player employing such as strategy as a *corrector player*. A corrector player believes that she knows her opponent's optimal strategy. When her opponent diverges from that strategy, the corrector player corrects this mistake (at her own expense).

In addition to finding optimal strategies for Angel and Demon, we also study the properties of the expected game length, with respect to the graph type, size, starting position of the tokens, and who moves first.

Acknowledgments

I would like to thank Dr. Andrew Beveridge, Dr. Alicia Johnson, Dr. Mike Weimerskirch and Manon Gentil for all of their help, patience and support in the production of this paper.

Chapter 2

Prerequisites

In this chapter we provide the background necessary to understand the main results of the paper. For a more complete introduction to probability theory and Markov chains, or graph theory please refer to [Kemeny and Snell (1976)] or [West (2000)], respectively.

2.1 Sets

We define a **set** as an arbitrary collection of objects, called **elements.** If a set B contains some but not necessarily all of the elements of another set A, we call B a subset of A and denote this relationship by, $B \subseteq A$. If B and A contain the same elements we say A *equals* B, or $A = B$. Finally, if $B \subseteq A$ and $B \neq A$ then we say B is a *proper subset* of A, or $B \subset A$.

In studying the process of a game of chance, we are naturally concerned with the possible outcomes of plausible events in the game. We shall denote the set of all *logically possible outcomes* (logical possibilities) after some event by U. We refer to such a set as a *possibility space.*

Example 2.1.1 After flipping a coin the possibility space is $U = \{T, H\}$. After flipping three coins the possibility space is

$$U = \{HHH, HHT, HTH, THH, HTT, THT, TTH, TTT\}.$$

$\diamondsuit$

We say that, if $u_1, u_2 \in U$:

- $\overline{u}_1$ (read: not u_1) is true if and only if u_1 is false.

- $u_1 \vee u_2$ (read: u_1 or u_2) is true if and only if either u_1 is true, u_2 is true or both u_1 and u_2 are true.
- $u_1 \wedge u_2$ (read: u_1 and u_2) is true if and only if both u_1 and u_2 are true.

2.2 Basics of Probability

2.2.1 Probability Measures

In defining a probability measure we let $S = \{s_1, s_2, s_3, \ldots, s_n\}$ be a set of logical possibilities. We obtain the probability measure for S by defining a function $w : S \rightarrow [0,1]$ such that $\sum_{i=1}^{n} w(s_i) = 1$. If $M \subseteq S$ then we define its measure, $m(M)$, as $\sum_{s_r \in M} w(s_r)$.

Theorem 2.2.1 *With any probability measure on a set U, the following (Kolmogorov) properties must hold:*

- *If $M \subseteq U$ then $0 \leq m(M) \leq 1$.*
- *If $M, N \subseteq U$ and $M \cap N = \emptyset$ then $m(M \cup N) = m(M) + m(N)$.*
- *If $M \subseteq U$ then $m(\overline{M}) = 1 - m(M)$.*

The proof of this theorem follows directly from the definition of a probability measure. For simplicity of notation, we will denote the measure function $m(X)$, by $Pr[X]$. We refer to $Pr[X]$ as the probability of X.

Example 2.2.2 Consider the set

$$U = \{HHH, HHT, HTH, THH, HTT, THT, TTH, TTT\}$$

containing the logically possible outcomes after each toss of a fair coin flipped 3 times. Since the coin is fair we know that for any $S \subseteq U$, if $s \in S$:

- $\Pr[s] = \frac{1}{|U|} = \frac{1}{2^3} = \frac{1}{8}$;
- $\Pr[S] = \frac{|S|}{|U|} = \frac{|S|}{8}$;
- $\Pr[\overline{S}] = 1 - \frac{|S|}{|U|} = \frac{8-|S|}{8}$.

$\diamond$

Example 2.2.3 Suppose that a jar has 5 red (R) marbles and 4 blue (B) marbles from which a blind individual will choose 2 marbles, one at a time, without replacement. The set of logically possible outcomes per reach is $U = \{R, B\}$. Similarly the set of possible outcomes after 2 reaches is $U = \{(R,R),(R,B),(B,R),(B,B)\}$. We deduce that:

- $\Pr[(R,B) \vee (B,B)] = \Pr[(R,B)] + \Pr[(B,B)] = \frac{5}{9} \cdot \frac{4}{8} + \frac{4}{9} \cdot \frac{3}{8} = \frac{32}{72}$;
- $\Pr[(R,B) \wedge (R,R)] = 0$;
- If $A = (R,B) \vee (B,B)$ then $\Pr[\overline{A}] = 1 - [\Pr[(R,B)] + \Pr[(B,B)]] = 1 - \frac{32}{72} = \frac{40}{72}$;

◇

In many experiments (or systems) we often encounter an infinite number of possible outcomes. As the following example demonstrates, the set of logical possibilities U need not be finite, so long as we choose an appropriate measure function.

Example 2.2.4 Consider an experiment where we are interested in the number of times we must roll a die before we obtain a 2. In this case, $U = \{1,2,3,\dots\}$. For $t \in U$, $\Pr[t] = \left(\frac{5}{6}\right)^{t-1}\frac{1}{6}$. Since

$$\sum_{t=1}^{\infty}\left(\frac{5}{6}\right)^{t-1}\frac{1}{6} = \frac{1}{6}\sum_{t=0}^{\infty}\left(\frac{5}{6}\right)^{t} = \frac{1}{6}\left(\frac{1}{1-\frac{5}{6}}\right) = 1$$

we know the probability space is well-defined. ◇

2.2.2 Conditional Probability

We are often concerned about the probability of some event A occurring, given that some other event B has occurred. We call this the *conditional probability* of A given B, denoted $\Pr[A|B]$. We define

$$\Pr[A|B] = \frac{\Pr[A \wedge B]}{\Pr[B]}.$$

A and B are *independent events* if $\Pr[A|B] = P[A]$, or equivalently

$$\Pr[A \wedge B] = \Pr[A]\Pr[B].$$

Example 2.2.5 Suppose Al, Bill and Carol each roll a fair die. Define the events

$$\begin{aligned} A &= \text{Al rolls a 1} \\ B &= \text{Bill rolls a 1} \\ C &= \text{Carol rolls a 1.} \end{aligned}$$

Clearly $\Pr[A] = \Pr[B] = \Pr[C] = \frac{1}{6}$ and $\Pr[A \wedge B] = \Pr[A]\Pr[B] = \frac{1}{36}$. With this knowledge we may find the probability of A occurring given that B has occurred.

$$\Pr[A|B] = \frac{\Pr[A \wedge B]}{\Pr[B]} = \frac{\frac{1}{36}}{\frac{1}{6}} = \frac{1}{6}.$$

◇

In this example, $\Pr[A|B]\Pr[B] = \Pr[A]P[B]$, and so A and B are independent.

Example 2.2.6 Let us consider Example 2.2.3 once again. Let A be the event that a red marble is chosen first, and B be the event that a blue marble is chosen second. Then

$$\Pr[B|A] = \frac{\Pr[A \wedge B]}{\Pr[A]} = \frac{\frac{5}{9}\frac{4}{8}}{\frac{5}{9}} = \frac{4}{8},$$

but

$$\Pr[B] = \Pr[B|A]\Pr[A] + \Pr[B|\overline{A}]\Pr[\overline{A}] = \frac{4}{8}\frac{5}{9} + \frac{3}{8}\frac{4}{9} = \frac{4}{9}.$$

Hence the events A and B are *dependent.* ◇

2.2.3 Expected Value and Variance

Thus far we have assigned probabilities to events in a finite or countably infinite set. At times, however, it may be desirable to redefine a set of outcomes by assigning numerical values to events. We shall refer to such a funtion as a *random variable.*

Example 2.2.7 Suppose that we pick 5 marbles out of a jar containing (no) red and blue marbles. Let $U = \{yes/no, yes/no, yes/no, yes/no, yes/no\}$, where yes in the n^{th} position, corresponds to the fact that we picked a red (blue) marble on the n^{th} pick. If we are only interested in the number of red marbles then we may define a random variable, ϕ, which maps each element of U to some element in $\{0,1,2,3,4,5\}$. For example

$$\phi : (yes, yes, no, yes, yes) \mapsto 4.$$

$\diamondsuit$

We refer to random variables that map to a countable number of elements, such as the one in the previous example, as *discrete random variables*. Naturally, we may be interested in knowing the average value of a random variable. In order to do this we must consider a weighted average which, of course, depends on the probability measure defined. We call this weighted average the *expected value*. For discrete random variables we find the expected value is

$$E[U] = \sum_{u \in U} \phi(u) \Pr[u].$$

One should always recall when working with expected values that they are not necessarily the most likely value of the random variable, but the weighted average. For example, if a random variable $\phi : U \to \{-1, 1\}$, and $\Pr[-1] = \Pr[1] = \frac{1}{2}$, the expected value is $E[\phi] = \frac{1}{2}(-1) + \frac{1}{2}(1) = 0$.

The following properties are easily derived:

$$\begin{aligned} \mathrm{E}[X + c] &= \mathrm{E}[X] + c, \\ \mathrm{E}[X + Y] &= \mathrm{E}[X] + \mathrm{E}[Y], \\ \mathrm{E}[cX] &= c\mathrm{E}[x]. \end{aligned}$$

We refer to these properties as consequences of the *linearity of expectation*.

2.3 Graphs

We define a graph G as a collection of vertices $V(G)$ and a collection of edges $E(G)$, where an edge is a pair of vertices. We say an edge e is *incident* with a vertices v, w if $e = \{v, w\}$. We say v and w are *adjacent* if there is an edge $e = \{v, w\}$. The degree of v, denoted $\deg(v)$, is the number of vertices that are adjacent to it.

A *path* P_{n+1} is a graph with $V(P_{n+1}) = \{0, \ldots, n\}$ and $E(P_{n+1}) = \{\{k, k+1\} \mid k \in \{0, \ldots, n-1\}\}$. The *length* of a path is the number of edges. The *order* of a graph is the number of vertices. We shall denote paths of order $i+1$ by P_{i+1}.

Figure 2.1 shows a path of order 6 and length 5.

P_6:

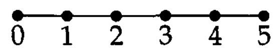

Figure 2.1: P_6 with 6 vertices and 5 edges.

Let P_i and P_j be paths such that $V(P_i) = \{v_1, \ldots, v_i\}$ and $V(P_j) = \{w_1, \ldots, w_j\}$. P_i and P_j are *internally disjoint* if $\{v_2, \ldots v_{i-1}\} \cap \{w_2 \ldots w_{j-1}\} = \emptyset$. In other words, P_i and P_j contain no vertices in common, except for maybe the first and last ones in their sequences. A *cycle* is the union of two internally disjoint paths with the same first and last vertices. In other words, a cycle is a graph in which exactly two distinct, internally disjoint paths exist from any vertex to another. These paths are *subgraphs* of the cycle. More generally a graph $H = (V(H), E(H))$ is a subgraph of $G = (V(G), E(G))$ when $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. A cycle of order i, is denoted by C_i. Figure 2.2 shows C_5.

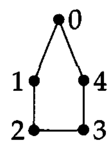

Figure 2.2: C_5: a cycle of order 5

A graph is *acyclic* if it does not contain a cycle as a subgraph. A graph G is connected, if for any vertices $v, w \in V(G)$, G contains a a path from v to w as a subgraph. A connected acyclic graph is a *tree*. Figure 2.3 shows a tree on 7 vertices.

In a *directed graph* G, $E(G)$ consists of arcs, (u, v). We denote an arc (u, v) on pictorial representations of graphs by an arrow pointing from u to v. If we are moving along edges of a directed graph, then we may only travel from u to v if there exists an arc (u, v). We define the distance between the vertices u, v, $d(u, v)$, as the minimum number of edges we must traverse in order to travel from u to v.
The *out-degree* of a vertex v in a directed graph is the number of arcs in $E(G)$ of the form (v, u), where $u \in V(G)$. On a directed graph, a *leaf* is a vertex with out-degree 1. On an undirected graph a leaf is a vertex with only one neighbor.

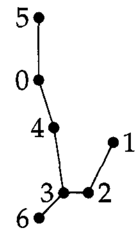

Figure 2.3: A tree of order 7

2.4 Markov Chains

Graphs and probability intersect in the study of random walks on graphs. In a simple random walk on a graph, at each time step we move from the current vertex to a randomly chosen neighbor. Random walks on graphs are more formally known as Markov chains. As explained in Coppersmith et al. (1993a), Markov chains have many applications: electrical network theory, estimation of measurements given by approximate measurements, finding the volume of a convex body, and on-line algorithms.

A *stochastic process* is a non-deterministic sequence of random variables. A *Markov chain* (or random walk) is a stochastic process where the i^{th} state depends only on the $(i-1)^{th}$ state. In particular, a Markov chain is a sequence of *random variables* $X_1, X_2 \ldots$, which each have possible outcomes $\{x_1, \ldots, x_i, \ldots\}$, such that

$$\Pr[X_n = x_n \mid X_1 = x_1, \ldots, X_{n-1} = x_{n-1}] = \Pr[X_n = x_n \mid X_{n-1} = x_{n-1}].$$

We define the transition probability as

$$p_{i,j} = \Pr[X_n = x_j \mid X_{n-1} = x_i]$$

and note that $p_{i,j}$ is independent of n. The *transition matrix* P of a Markov chain has $p_{i,j}$ as the entry in the i^{th} row and j^{th} column. In particular, if all edges are equally likely to be traversed,

$$p_{i,j} = \begin{cases} \frac{1}{\deg(i)} & \text{if } \{i,j\} \in E(G) \\ 0 & \text{otherwise.} \end{cases}$$

Example 2.4.1 Suppose that Al is walking on the graph C_4 below, starting on vertex 0. At each time step, Al will move to a randomly chosen adjacent

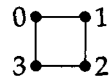

Figure 2.4: C_4: Four Cycle

vertex. The transition matrix at the first step is

$$P = \begin{pmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{pmatrix}.$$

where $P_{i,j}$ is the probability of moving from vertex i to vertex j in one step. It follows that the transition matrix after t total steps is

$$P^t = \begin{pmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{pmatrix}^t.$$

Here, $P^t_{i,j}$ is the probability of moving from vertex i to vertex j in exactly t steps. $\diamondsuit$

We define an *absorbing state* on a graph G to be a vertex that ends a random walk. In other words, for aborbing state v_i, the transition probability $p_{i,i} = 1$, and $p_{i,j} = 0$ for $j \neq i$. We naturally refer to Markov chains with an absorbing state as *absorbing Markov chains*. These will be of particular concern for the problems we will be considering.

Example 2.4.2 Let P_5 be a graph with absorbing state 0. Figure 2.5 demonstrates the potential motion of a token taking a random walk on P_5.

$\diamondsuit$

The expected time before a token starting on j reaches some vertex v, is called the *hitting time*, and is denoted by $H(j, v)$.

2.4.1 Coupling

Coupling is a tool for relating two distinct random variables (or processes). We give an example of how to employ a coupling in order to force the steps

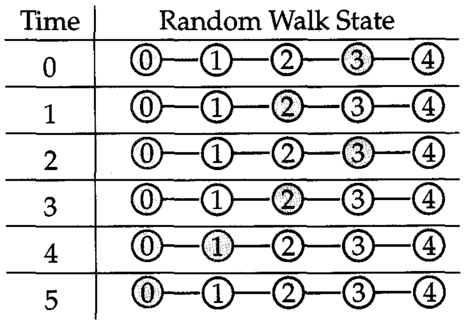

Figure 2.5: Trajectory on P_5 with absorbing state 0

of 2 distinct random walks to correspond. In essence, we use the same random number generator for two different processes so that they evolve in tandem.

Example 2.4.3 We consider a random walk on two distinct paths. We let a fair coin determine the direction of the token step. For our coupling, we let one coin determine the motion of both tokens. If tails, the token moves left, if heads the token moves right. Suppose we obtain the the following outcomes for 3 coin flips: T, T, H.

Using these coin flip outcomes, random walks on P_5 with tokens starting at vertices 1 and 3 proceed as shown in Figure 2.6 below.

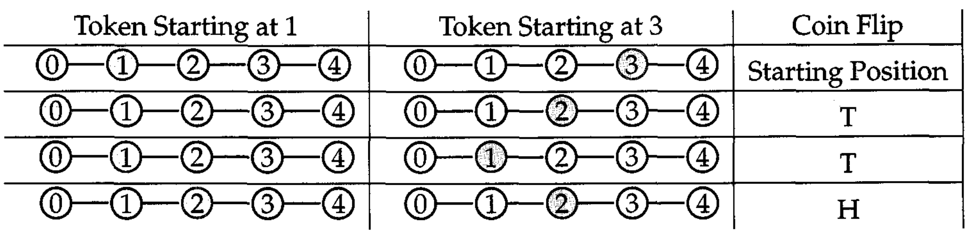

Figure 2.6: Aligning tokens

Note that, as seen on step 3, once the green and blue tokens *align*, they will remain aligned under the given coupling. $\diamondsuit$

2.5 Combinatorial Games

A game with positions, in which two or more players take turns altering the position, is considered a combinatorial game if all players have *perfect information*. A player has perfect information if he/she always knows the position of the game, and how players can alter it. Connect Four, Chess, Checkers, Backgammon and Go are popular examples of combinatorial games. Games such as Spades, Hearts and Poker, are not combinatorial games, because the player does not know the cards (i.e. position) that the other players have.

We say that a game is solved when we determine a strategy for each player that maximizes the probability of achieving some predetermined game state, referred to as a winning condition. Games can achieve a large degree of complexity and hence we require a set of techniques that allows us to work with their complexity.

2.5.1 Strategy Stealing

Strategy stealing is often employed in combinatorial game theory in order to demonstrate that a player does not have a winning strategy. When employing strategy stealing we allow a player to *steal* the move of another player, not necessarily playing the same game. We consider the game tic-tac-toe on a 3-by-3 board as an example.

Example 2.5.1 Anyone who has played tic-tac-toe eventually realizes that the first player to move has a notable advantage. We prove, via an employment of strategy stealing, that the player who moves second cannot have a winning strategy.

Assume for the sake of contradiction that the player who moves second, P_2, has a winning strategy, call it S. We have the first player, P_1, choose his first move at random. After P_2's move, P_1 steals S, pretending that P_2's first move was the first move of the game. If ever S dictates that he should move where his first move took place, P_1 should move randomly. P_1 plays as though he is P_2, so this modified strategy is a winning one. This is a contradiction. $\diamondsuit$

Chapter 3

Simultaneous Random Walks

In this chapter we discuss previous work on simultaneous random walks, and introduce the game we will study.

3.1 Motivation

Our work builds on results in *Collisions among Random Walks on a Graph* by Coppersmith, Tetali and Winkler (Coppersmith et al. (1993b)). The original motivation for their paper comes from self-stabilizing token management schemes. Protocols for distributed networks are *self-stabilizing* if given any unstable, or irregular, state, they eventually return to a stabilized state allowing for regular operation. In such processor self-stabilizing token management schemes, we only allow for one processor to be active at a time. When this processor's operations are complete, it passes an ownership token along to another processor. The token represents some abstract object that is sent from processor to processor. Note that we can consider each processor as a vertex on a graph where an edge exists from vertex (processor) v to vertex (processor) w if a token can be sent from v to w.

Suppose that our distributed network enters an illegal state when there is more than one token in the system. In Israeli and Jalfon (1990), the authors suggest randomly passing tokens between processors until they collide. On such a collision, the tokens merge, bringing the system back into a legal state. We say that equilibrium is achieved when only one token remains. The processors in this self-stabilizing token management scheme are not synchronized. Instead, which processor is activated is determined by an entity, which we will refer to as a *player*.

Previous work by Coppersmith et al. (1993b) and Tetali and Winkler

(1993) considers the case where one player manages the activation of processors. We expand on the results of these papers by considering a game version of the token management scheme. In our game, two players alternate turns until the two tokens collide.

Unless stated otherwise, we assume that a processor has an equal probability of sending a token to any of the neighboring processors. Hence, in this paper we are studying multiple, simultaneous, Markov Chains on a graph.

3.2 The Single Player Game

In Coppersmith et al. (1993b), three types of regulatory players are studied.

- **Angel (A):** Angel plays to minimize the expected game length.
- **Demon (D):** Demon plays to maximize the expected game length.
- **Random (R):** Randomly chooses which token to move at each step.

We assume that Angel and Demon are both playing *optimally*, so that they are respectively minimizing and maximizing the expected game length. Coppersmith et al. (1993b) and Tetali and Winkler (1993) study single player games where one player controls every move. We refer to simultaneous random walks, where the same player chooses which token moves at each time step, as *single player games*.

In this paper we only study games where two tokens are placed on a graph. If the two tokens, call them τ_L and τ_R, are on vertices i, j, respectively, then we say we are in game state (i,j). We denote the game where player S regulates the game on the graph G with initial token position (i,j), by

$$\mathcal{G} = (G, S, i, j).$$

The *meeting time*

$$M_S^{\mathcal{G}}(i,j)$$

is the expected number of activations before the tokens meet when following strategy S. When it is clear which graph we are referring to, we use the notation $M_S(i,j)$ instead of $M_S^{\mathcal{G}}(i,j)$.

Known Results for Single Player Games

Equipped with the necessary understanding of stochastic processes, we summarize some of the results in Coppersmith et al. (1993b) and Tetali and Winkler (1993).

As shown in Tetali and Winkler (1993), the expected meeting time of the tokens is bounded for the single player Angel, Demon and Random games.

Theorem 3.2.1 (Tetali and Winkler (1993)) *Let G be a graph of order n. Let*

$$\begin{aligned} M_D^G(n) &= \max_{u,v \in V(G)} M_D^G(u,v), \\ M_A^G(n) &= \max_{u,v \in V(G)} M_A^G(u,v), \\ M_R^G(n) &= \max_{u,v \in V(G)} M_R^G(u,v), \end{aligned}$$

and

$$\begin{aligned} M_D(n) &= \max_{G:|G|=n} M_D^G(n), \\ M_A(n) &= \max_{G:|G|=n} M_A^G(n), \\ M_R(n) &= \max_{G:|G|=n} M_R^G(n). \end{aligned}$$

In other words, $M_D(n)$ is the maximum meeting time over all n-vertex graphs and all starting positions, when, at each time step, Demon is choosing the token that moves; $M_A(n)$ and $M_R(n)$ are similarly defined.
Then

$$\begin{aligned} &M_D(n) = \left(\frac{4}{27} + o(1)\right) n^3; \\ &\left(\frac{1}{27} + o(1)\right) n^3 \le M_A(n) \le \left(\frac{4}{27} + o(1)\right) n^3; \\ &\left(\frac{1}{27} + o(1)\right) n^3 \le M_R(n) \le \left(\frac{4}{27} + o(1)\right) n^3. \end{aligned}$$

Hence, no strategy exists for Demon that will forever keep the tokens from colliding.

A player's strategy is *pure* if at each turn the choice of token depends only on the current locations of the tokens. As observed in Coppersmith et al. (1993b), a pure strategy is equivalent to a *tournament* on the vertices of the graph; at each pair of vertices, the winner is the vertex chosen by the player. Both Angel and Demon have pure strategies in their single player game.

Lemma 3.2.2 (Coppersmith et al. (1993b)) *In the single player game, on any graph G the Demon and the Angel both have a pure optimal strategy.*

A strategy is a *degree strategy* if whenever the tokens are on vertices i and j, and $\deg(i) < \deg(j)$, the token on j is moved.

Theorem 3.2.3 (Coppersmith et al. (1993b)) *On a path, a strategy for Demon, in a single player game, is optimal if and only if it is a degree strategy.*

We replicate the proof in Coppersmith et al. (1993b) because it is a nice example of random walk coupling.

Proof. Let P_n be a path with, $V(P_n) = \{v_0, \ldots, v_{n-1}\}$. As a pair of tokens moves within G, let us imagine a single *distance token* moving on a second copy, P'_n, of P_n according the following rule: if the tokens on G are at vertices v_i and v_j, then the location v'_k of the distance token is given by $k = |i - j|$.

If the demon follows a degree strategy, then the distance token takes a uniform random walk on P'_n, which has an absorbing state v'_0. Hence, under a degree strategy, the expected game length is $H_{P'_n}(v'_k, v'_0)$, the expected number of steps before a token starting on v'_k reaches v'_0. However, if at any game state (v_i, v_{n-1}) or (v_0, v_i), where $0 < i < n-1$, the demon has probability $p > 0$ of moving the token on the leaf, then the probabilities along the edges leading from the corresponding vertex on G' are skewed toward v'_0, decreasing the expected time to finish. ∎

This optimal Demon strategy, for the path, only prescribes what happens when one token is on a leaf. Otherwise, Demon may move either token. However, the optimal Angel strategy for a single player game is more restrictive. A similar argument proves that moving a token off a leaf is beneficial for Angel. Additionally, an intricate argument found in Coppersmith et al. (1993b) (which we will not provide here) gives us the following:

Theorem 3.2.4 (Coppersmith et al. (1993b)) *Angel need never move a token off the center.*

In a hitting time strategy, a token on vertex x is chosen over a token on vertex y when $H(x,y) < H(y,x)$. When $H(x,y) = H(y,x)$ a token is arbitrarily chosen.

Theorem 3.2.5 (Tetali and Winkler (1993)) *The hitting time strategy is optimal on a tree for the Angel.*

3.3 The Two Player Game

We study a two player process, in which the players alternately choose which token to move. We denote a two player game on a graph G by

$$\mathcal{G} = (G, S_1, S_2, i, j).$$

The player strategies in $\mathcal{G}$ are S_1, S_2, and S_1 moves first; the initial token position of $\mathcal{G}$ is (i, j). We denote the expected game length by

$$M^G_{S_1,S_2}(i, j).$$

The abbreviated notation $M_{S_1,S_2}(i, j)$ is used when the graph under consideration is clear from the context.

We define a *trajectory* of a game as a potential course of the game determined by the initial positions, the player strategies, and how the tokens move at each time step.

Example 3.3.1 Let P_5 be a path with $V(P_5) = \{0, 1, 2, 3, 4\}$. Suppose that S_1 is a player that always moves the token on the vertex with smaller index; vice versa for S_2. Figure 3.1 illustrates a potential trajectory of the game $(P_5, S_1, S_2, 1, 3)$, where coin flips are used to determine how the tokens step at each step (either left(L) or right (R)). $\diamondsuit$

3.3.1 A Simple Lemma about Leaves

For any graph G and strategies S_1, S_2, let

$$M^G_{S_1,S_2}(\overline{u}, v) = \frac{1}{\deg(u)} \sum_{w \in N(u)} M^G_{S_2,S_1}(w, v)$$

where $N(u)$ is the set of neighbors of u; define $M_{S_1,S_2}(u, \overline{v})$ analogously. The following simple Lemma proves useful for the following sections.

Lemma 3.3.2 *Let u be a leaf and v its unique neighbor, that is not a leaf, on some graph G. In the game (G, A, D, u, v) Angel moves the token at u. In the game (G, D, A, u, v) Demon moves the token at v.*

Game State	Controlling Player	t	coin flip outcome
$v_0 - v_1 - v_2 - v_3 - v_4$	S_1	0	L
$v_0 - v_1 - v_2 - v_3 - v_4$	S_2	1	R
$v_0 - v_1 - v_2 - v_3 - v_4$	S_1	2	R
$v_0 - v_1 - v_2 - v_3 - v_4$	S_2	3	L
$v_0 - v_1 - v_2 - v_3 - v_4$	S_1	4	

Figure 3.1: A trajectory of $(P_5, S_1, S_2, 1, 3)$; L (R) moves the selected token left (right); S_1 (S_2) always chooses the token at the vertex with a lower (higher) index.

Proof.

$$M_{D,A}(u,v) = 1 + \max \begin{cases} M_{A,D}(v,v) = 0, \\ M_{A,D}(u,\overline{v}) \geq 1. \end{cases}$$

Hence $M_{D,A}(u,v) = 1 + M_{A,D}(u,\overline{v})$, meaning that Demon moves the token at v. It similarly follows that

$$M_{A,D}(u,v) = 1 + \min \begin{cases} M_{D,A}(v,v) = 0, \\ M_{D,A}(u,\overline{v}) \geq 1. \end{cases}$$

Hence $M_{A,D}(0,1) = 1$, meaning that Demon moves the token at u. ∎

3.3.2 Pure Optimal Strategies

On any graph, Angel and Demon have pure optimal strategies in our two player game. In other words, there is exists an optimal strategy, such that each move depends only on the current state of the game, and a particular token is chosen with probability 1. Although this may seem evident because pure optimal strategies exist in the single player game, this statement requires proof.

Lemma 3.3.3 *For any game on some graph G, Angel (A) has a pure optimal strategy.*

Proof. This proof closely follows the proof of Lemma 3.2.2 given in Coppersmith et al. (1993b). Fix an optimal Demon strategy D. Let $S(i,j)$ be a strategy minimizing $M_{S(i,j),D}(i,j)$. Define a tournament T by letting i be the winner over j when given a starting game state (i,j), $S(i,j)$ moves the token at i; similarly for j. If either token may be moved we assign a winner arbitrarily. We claim that with the pure strategy A_p, corresponding to the tournament T, $M_{A,D}(i,j) = M_{A_p,D}(i,j)$, for all i,j.

Assume for the sake of contradiction that $M_{A,D}(i,j) < M_{A_p,D}(i,j)$ for some vertices i,j. Let

$$\alpha = \max\{M_{A_p,D}(i,j) - M_{A,D}(i,j) \mid i,j \in V(G)\}.$$

Of all pairs of vertices x,y such that $\alpha = M_{A_p,D}(x,y) - M_{A,D}(x,y)$, choose one with the minimum distance. Assume without loss of generality that x beats y in T and that with tokens starting at x and y, $S(x,y)$ moves x with probability $p > 0$. We claim that

$$M_{D,A_p}(\overline{x},y) - M_{D,A}(\overline{x},y) < \alpha. \tag{3.1}$$

If $d(x,y) = 1$, then the probability that the game (G, A_p, D, x, y) ends after one move is $1/\deg(x) > 0$. If the game has not ended, then Demon takes his turn. Afterwards, we are in the game (G, A_p, D, u, v) for certain pairs $u, v \in V$. For each such game, we have $M_{A_p,D}(u,v) - M_{A,D}(u,v) \le \alpha$.

Let $Z_{u,v}$ denote the event that the tokens did not collide after the first move of A_p and that the tokens are at u,v after Demon's next move. Note that $\Pr[\vee_{u,v\in V} Z_{u,v}] = \sum_{u,v\in V} \Pr[Z_{u,v}] = 1 - 1/\deg(x)$. We have

$$\begin{aligned} M_{D,A_p}(\overline{x},y) - M_{D,A}(\overline{x},y) &= \sum_{u,v\in V} (M_{A_p,D}(u,v) - M_{D,A}(u,v)) \Pr[Z_{u,v}] \\ &\le \alpha(1 - 1/\deg(x)) < \alpha, \end{aligned}$$

so equation (3.1) holds in this case.

The case $d(x,y) > 1$ is similar. In this case, the game cannot complete after the first move of A_p. There exists a pair $x', y' \in V$ such that $\Pr[Z_{x',y'}] > 0$ and $d(x',y') < d(x,y)$. Indeed, the token moved in each round could move along the shortest path towards the other. For this pair, we have $M_{A_p,D}(x',y') - M_{A,D}(x',y') < \alpha$ since x,y were taken to be at minimal distance. Let $Z_{u,v}$ denote the event that the tokens are at u,v these two moves. We have

$$\begin{aligned} & M_{D,A_p}(\overline{x},y) - M_{D,A}(\overline{x},y) \\ &= \sum_{u,v\in V} (M_{A_p,D}(u,v) - M_{D,A}(u,v)) \Pr[Z_{u,v}] \\ &\le (M_{A_p,D}(u,v) - M_{D,A}(u,v)) \Pr[Z_{x',y'}] + \alpha(1 - \Pr[Z_{x',y'}]) < \alpha. \end{aligned}$$

so equation (3.1) holds in this case as well.

Recall that x beats y in T and that with tokens starting at x and y, $S(x,y)$ moves x with probability $p > 0$. We have

$$M_{S(x,y),D}(x,y) = 1 + pM_{D,A}(\overline{x},y) + (1-p)M_{D,A}(x,\overline{y})$$

because $S(x,y)$ is supposed to be optimal at x,y. Additionally, we must have that

$$M_{S(x,y),D}(x,y) = 1 + M_{D,A}(\overline{x},y)$$

since otherwise moving y would be optimal.

Hence,

$$\begin{aligned}
M_{S(x,y),D}(x,y) &= 1 + M_{D,A}(\overline{x},y) \\
&< 1 + M_{D,A_p}(\overline{x},y) + \alpha \\
&= M_{A_p,D}(x,y) + \alpha \\
&= M_{A,D}(x,y) \\
&= M_{S(x,y),D}(x,y),
\end{aligned}$$

where the inequality is due to equation (3.1). However, this strict inequality gives a contradiction, so it must hold that $M_{A,D}(x,y) = M_{A_p,D}(x,y)$. Therefore, A_p is a pure optimal strategy. ∎

Lemma 3.3.4 *For any game on some graph G, Demon (D) has a pure optimal strategy.*

Proof. The proof is analogous to the previous one. We let $S(i,j)$ maximize $M_{D,S(i,j)}(i,j)$, and assuming that $M_{D,A}(x,y) > M_{D_p,A}(x,y)$ for some vertices x,y. ∎

Chapter 4

Results Using Naive Methods

In this chapter we use some naive methods to prove our first results. We start with optimal Angel and Demon strategies on the star. This simple example gives a first glimpse into the characteristics of optimal strategies. We also find optimal strategies on small paths by solving systems of linear equations. This method is intractable for larger paths.

While reading the chapter note that there are two distinct games to analyze on a path or star. This is a direct consequence of the bipartite nature of these graphs. In one game, Angel controls the odd distances between tokens. In the other game, Demon controls the odd distances.

4.1 Optimal Play on the Star

Star graphs are trees that contain exactly one non-leaf vertex. Figure 4.1 provides an example of a star of order 9. The optimal strategies for Angel and Demon are easy to identify.

Theorem 4.1.1 *On a star graph, when a choice exists, Angel should always move the token on a leaf, and Demon should always move the token not on a leaf.*

Proof. There are two cases to consider. The case where only one token is on a leaf, and the case where both tokens are on a leaf. Lemma 3.3.2 gives us the optimal strategy in the first case, and the symmetry of positions gives us the optimal strategy in the second case. ∎

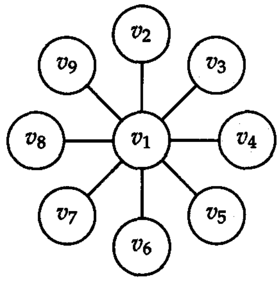

Figure 4.1: Star Graph of order 9

4.2 The Path on 3 Vertices

We will begin our exploration of our Angel versus Demon game on paths by considering P_3. We find optimal strategies for both Angel and Demon,

Figure 4.2: P_3, the path on 3 vertices.

and then use them to find the expected number of steps until the 2 tokens collide. We use these results to determine the *expected winner* of the game, by using the random game as a benchmark. Angel (Demon) wins if the expected game length of the two player game is less (greater) than the expected game length with only the Random player.

4.2.1 Random Game

In order to establish the "victory" thresholds we create a system of linear equations for the expected length of the random game. Observe that for any given graph this system is of the form

$$M_R(i,j) = 1 + \frac{1}{2}\left(M_R(\bar{\imath},j) + M_R(i,\bar{\jmath})\right).$$

Expanding and using the symmetry of the path, we find that

$$M_R(0,2) = 1 + \frac{1}{2}\left(M_R(1,2) + M_R(0,1)\right) = 1 + M_R(1,2)$$
$$M_R(1,2) = 1 + \frac{1}{2}\left(\frac{1}{2}M_R(0,2) + \frac{1}{2}M_R(2,2)\right) + \frac{1}{2}M_R(1,1) = 1 + \frac{1}{4}M_R(0,2).$$

Solving this system of equations gives

$$M_R(0,2) = 4 \text{ and } M_R(0,1) = M_R(1,2) = 3,$$

giving us the victory thresholds for each starting position.

4.2.2 Optimal Play: Demon Controls Odd Distances Between Tokens

Now we consider the expected game length when Angel and Demon are employing their optimal strategies. Identifying an optimal strategy is easy in such a small system. As with the random game, we will exploit the symmetry of the graph in order to simplify the system of equations.

On a path, the same player always moves when the tokens are an odd distance apart. In this section, we study the games, $(P_3, A, D, 0, 2)$ and $(P_n, D, A, 0, 1) \equiv (P_n, D, A, 1, 2)$, in which Demon moves when the tokens are an odd distance apart.

Notice that in positions such as $(0,2)$ it does not matter which token is moved and that positions $(0,1)$ and $(1,2)$ are equivalent.
Therefore,

$$M_{A,D}(0,2) = 1 + \frac{1}{2}M_{D,A}(1,2) + \frac{1}{2}M_{D,A}(0,1) = 1 + M_{D,A}(0,1),$$

and, by Lemma 3.3.2,

$$\begin{array}{rcccl} M_{A,D}(0,1) & = & M_{A,D}(1,2) & = & 1 \\ M_{D,A}(0,1) & = & M_{D,A}(1,2) & = & 1 + \frac{1}{2}M_{A,D}(0,2). \end{array}$$

Solving for the system of equations,

$$M_{A,D}(0,2) = 1 + M_{D,A}(0,1)$$
$$M_{D,A}(0,1) = 1 + \frac{1}{2}M_{A,D}(0,2)$$

we find that $M_{D,A}(0,1) = M_{D,A}(1,2) = 3$ and $M_{A,D}(0,2) = 4$.

4.2.3 Optimal Play: Angel Controls Odd Distances Between Tokens

It remains to study the games in which Angel moves when the tokens are an odd distance apart. We proceed as before, first noting the symmetry of the path which yields that

$$M_{D,A}(0,2) = 1 + \frac{1}{2}M_{A,D}(0,1) + \frac{1}{2}\ M_{A,D}(1,2) = 1 + M_{A,D}(1,2).$$

Thus, we obtain the following system of equations

$$\begin{aligned} M_{D,A}(0,2) &= 1 + M_{A,D}(1,2) \\ M_{A,D}(0,1) &= 1 \end{aligned}$$

from which we find that $M_{A,D}(0,1) = M_{A,D}(1,2) = 1,\ M_{D,A}(0,2) = 2.$

4.2.4 Expected Winners

From our previous results we gather the expected winners of each game on the path, on 3 vertices.

$M_{A,D}(0,1)$	$M_{D,A}(0,1)$	$M_{A,D}(0,2)$	$M_{D,A}(0,2)$
A	=	=	A

Table 4.1: The winner for games on P_3.

4.3 The Path on 4 Vertices

Let us now consider the more complicated case of a path of length 4.

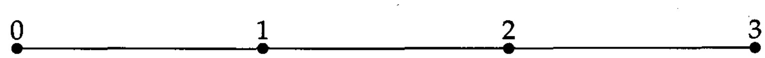

Figure 4.3: P_4, the path on 4 vertices

4.3.1 Random Game

Observing the symmetries in the given graph we find:

$$\begin{aligned}
M_R(0,3) &= 1 + M_R(0,2);\\
M_R(1,2) &= 1 + \frac{1}{2}M_R(0,2);\\
M_R(0,2) &= 1 + \frac{1}{2}M_R(1,2) + \frac{1}{4}M_R(0,1) + \frac{1}{4}M_R(0,3);\\
M_R(0,1) &= M_R(2,3) = 1 + \frac{1}{4}M_R(0,2).
\end{aligned}$$

Solving this system of equations yields

$$M_R(0,3) = \frac{39}{7},\ M_R(0,2) = \frac{32}{7},\ M_R(1,2) = \frac{23}{7},\ M_R(0,1) = \frac{15}{7}.$$

4.3.2 Optimal Play: Angel Controls Odd Distances Between Tokens

We now find the optimal strategies for Angel and Demon. Once again we establish a system of linear equations which will yield the expected game lengths. It clearly must hold that

$$\begin{aligned}
M_{A,D}(0,1) &= 1,\\
M_{A,D}(1,2) &= 1 + \frac{1}{2}M_{D,A}(0,2),\\
M_{A,D}(0,3) &= 1 + M_{D,A}(0,2).
\end{aligned}$$

It only remains to find

$$\max \begin{cases} \frac{1}{2}\left(M_{A,D}(0,1) + M_{A,D}(0,3)\right) \\ M_{A,D}(1,2) \end{cases} = M_{D,A}(0,2) - 1.$$

We claim that

$$M_{A,D}(1,2) = \frac{1}{2}\left(M_{A,D}(0,1) + M_{A,D}(0,3)\right).$$

In other words, it does not matter which token Demon moves at $(0,2)$. Using the symmetry of position $(1,2)$,

$$\begin{aligned} M_{A,D}(1,2) &= 1+\frac{1}{2}M_{D,A}(0,2) \\ &= \frac{1}{2}\left[2+M_{D,A}(0,2)\right] \\ &= \frac{1}{2}\left[M_{A,D}(0,1)+1+M_{D,A}(0,2)\right] \\ &= \frac{1}{2}\left[M_{A,D}(0,1)+M_{A,D}(0,3)\right]. \end{aligned}$$

Solving the system of linear equations

$$\begin{aligned} &M_{A,D}(0,3) = 1+M_{D,A}(0,2) \\ &M_{A,D}(1,2) = 1+\frac{1}{2}M_{D,A}(0,2) \\ &M_{D,A}(0,2) = 1+M_{A,D}(1,2) \\ &M_{A,D}(0,1) = 1 \end{aligned}$$

we find that

$$M_{A,D}(0,3)=5,\ M_{A,D}(1,2)=3,\ M_{D,A}(0,2)=4,\ M_{A,D}(0,1)=1.$$

4.3.3 Optimal Play: Demon Controls Odd Distances Between Tokens

We have already established in Lemma 3.3.2 that $M_{D,A}(0,1)=1+\frac{1}{2}M_{A,D}(0,2)$. Due to the symmetry of the path

$$\begin{aligned} M_{D,A}(0,3) &= 1+M_{A,D}(0,2) \\ M_{D,A}(1,2) &= 1+\frac{1}{2}M_{A,D}(0,2). \end{aligned}$$

It only remains to find the optimal move for $(P_4,A,D,0,2)$. We have

$$M_{A,D}(0,2)=1+\min\begin{cases} \frac{1}{2}\left(M_{D,A}(0,1)+M_{D,A}(0,3)\right) \\ M_{D,A}(1,2). \end{cases}$$

We claim that $\frac{1}{2}\left(M_{D,A}(0,1) + M_{D,A}(0,3)\right) > M_{D,A}(1,2)$. Indeed,

$$\begin{aligned} M_{D,A}(1,2) &= 1 + \frac{1}{2}M_{A,D}(0,2) < 1 + \frac{3}{2}M_{A,D}(0,2) \\ &= \frac{1}{2}\left[1 + \frac{1}{2}M_{A,D}(0,2) + 1 + M_{A,D}(0,2)\right] \\ &= \frac{1}{2}\left[M_{D,A}(0,1) + M_{D,A}(0,3)\right], \end{aligned}$$

and thus $M_{A,D}(0,2) = 1 + M_{D,A}(1,2)$. Solving for the system of linear equations

$$\begin{aligned} M_{D,A}(0,3) &= 1 + M_{A,D}(0,2) \\ M_{D,A}(1,2) &= 1 + \frac{1}{2}M_{A,D}(0,2) \\ M_{A,D}(0,2) &= 1 + M_{D,A}(1,2) \\ M_{D,A}(0,1) &= 1 + \frac{1}{2}M_{A,D}(0,2) \end{aligned}$$

yields that $M_{D,A}(0,3) = 5$, $M_{D,A}(1,2) = 3$, $M_{A,D}(0,2) = 4$, $M_{D,A}(0,1) = 3$.

4.3.4 Expected Outcomes

We summarize our results about the winner of the game for each initial position. Recall that Angel wins if the game finished faster (on average) than the random game, and that Demon wins if the game takes longer (on average) than the random game.

$M_{A,D}(0,1) = M_{A,D}(2,3)$	$M_{D,A}(0,2) = M_{D,A}(1,3)$	$M_{A,D}(1,2)$	$M_{A,D}(0,3)$
A	A	A	A

Table 4.2: Angel Controls Odd Token Distances

$M_{D,A}(0,1) = M_{D,A}(2,3)$	$M_{A,D}(0,2) = M_{A,D}(1,3)$	$M_{D,A}(1,2)$	$M_{D,A}(0,3)$
D	A	A	A

Table 4.3: Demon Controls Odd Token Distances

Resolving the min/max relations becomes quite challenging for P_n $n \geq 5$. Therefore, in Chapter 6 we turn to more sophisticated methods to identify optimal strategies for general paths.

Chapter 5

Lazy One-way Cycle

Establishing optimal strategies requires techniques from probability theory as well as combinatorial game theory. In particular, we will make use of random walk *coupling* and *strategy stealing*. In this chapter, we identify optimal 2-player game strategies on the *lazy one-way cycle* and the *biased cycle*. The proofs follow a similar structure as the proofs for the path, and so this chapter provides some intuition and familiarity before encountering the more convoluted proofs for paths.

A lazy one-way cycle is a *weighted* directed cycle, where for each vertex v, there exist unique arcs of the form (v,v), (v,w), (u,v), where v,w,u are distinct vertices. We define the weight on an edge as the probability of traversing it. Letting $\alpha \in [0,1]$, we designate the transition probabilities that make up the transition matrix as follows.

$$p_{i,j} = \begin{cases} \alpha & \text{if } j = i+1 \\ 1-\alpha & \text{if } i = i \\ 0 & \text{otherwise.} \end{cases}$$

Figure 5.1 shows such a lazy-one way cycle of order 6, where $\alpha = \frac{2}{3}$. We will always assume that a lazy one-way cycle G of order $n+1$, has

$$\begin{aligned} V(G) &= \{i \mid i \in \{0,\ldots,n\}\} \\ E(G) &= \{(i,(i+1) \bmod (n+1)) \mid i \in \{0,\ldots,n\}\} \cup \{(i,i) \mid i \in \{0,\ldots,n\}\}. \end{aligned}$$

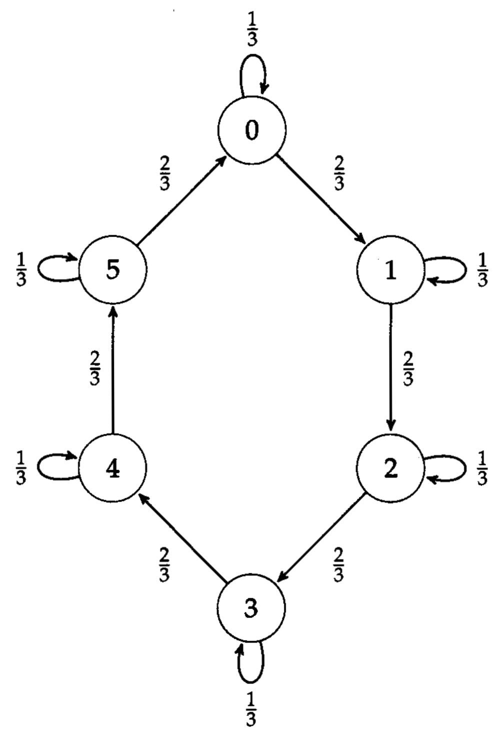

Figure 5.1: Lazy One-way Cycle of Order 6 with $\alpha = \frac{2}{3}$

5.1 Optimal Strategies for Angel and Demon

It does not take long to develop an intuition of the optimal strategies for both Angel and Demon. Angel should try to decrease the distance between the tokens and vice versa for Demon. Assume without loss of generality that the two tokens are placed on vertices 0 and j, on a graph G of order $n+1$, where $j \leq \lfloor \frac{n+1}{2} \rfloor$. When $n+1$ is even and $j = \frac{n+1}{2}$, the choice of tokens does not matter. In all other cases, our intuition is that A should move the token at 0 and D should move the token at j.

Let $\tilde{A}$ ($\tilde{D}$) denote this concrete strategy for Angel (Demon). We now provide a formal proof that $\tilde{A}$ and $\tilde{D}$ are the optimal strategies. We use a fruitful technique from combinatorial game theory. In order to establish a relationship between A and $\tilde{A}$, we introduce a third strategic player. The player serves as an intermediary to bridge the gap between A and $\tilde{A}$.

Let C_A (C_D) be the Angel (Demon) *corrector player*. We have C_A (C_D) play against A (D). C_A plays like $\tilde{D}$ unless Angel's pure strategy in some position deviates from $\tilde{A}$'s; C_A then "corrects" optimal Angel's move by moving the token that $\tilde{A}$ would have moved in the previous step. Similarly, C_D plays like $\tilde{A}$ unless Demon's move deviates from $\tilde{D}$'s, in which case it "corrects" Demon's move.

Lemma 5.1.1 *If G is a lazy one-way cycle of order $n+1$ then* $M_{\tilde{A},\tilde{D}}(i,j) \leq M_{A,D}(i,j)$.

Proof. Since

$$M_{A,C_A}(i,j) \leq M_{A,D}(i,j)$$

it suffices to show that

$$M_{\tilde{A},\tilde{D}}(i,j) = M_{A,C_A}(i,j).$$

Assume that $i = 0$ and $j \leq \lfloor \frac{n+1}{2} \rfloor$. We consider two cases.
Case 1: $j = 1$.
Considering optimal Angel versus optimal Demon, we have

$$\begin{aligned} M_{A,D}(0,1) &= 1 + \min \left\{ \begin{array}{c} (1-\alpha)M_{D,A}(0,1) \\ (1-\alpha)M_{D,A}(0,1) + \alpha M_{D,A}(0,2). \end{array} \right. \\ &= 1 + (1-\alpha)M_{D,A}(0,1), \end{aligned}$$

since $M_{D,A}(0,2) > 0$. For our concrete strategies we have

$$M_{\tilde{A},\tilde{D}}(0,1) = 1 + (1-\alpha)M_{\tilde{D},\tilde{A}}(0,1).$$

In other words, the optimal strategy A moves the token at i. This is the same token chosen by the concrete strategy $\tilde{A}$ in the same configuration.
Case 2: $j > 1$.
Let $\mathcal{G}_1 = (G, \tilde{A}, \tilde{D}, 0, j)$ and $\mathcal{G}_2 = (G, A, C_A, 0, j)$. We may assume that each token has its own random, predetermined, bit-string. We couple $\mathcal{G}_1$ and $\mathcal{G}_2$ by using the bit-string $B_1 = \{b_{1,1}, b_{1,2}, \ldots\}$ for the tokens on 0 and the bit-string $B_2 = \{b_{2,1}, b_{2,2}, \ldots\}$ for the tokens on j. With this coupling, the games play out identically unless there is deviation between $\tilde{A}$ and A. It follows from Case 1, that such a deviation does not occur when $j = 1$. Thus, the distance between the tokens at a deviation will be at least 2.

Suppose strategies A and $\tilde{A}$ move different tokens for game state $(0, j)$, $j > 1$. Note that the games cannot end after this move since $j - 1 > 1$. We claim that the games will once again have identical token positions one move after this deviation. Indeed, in the move following the deviation, C_A will move the token $\tilde{A}$ moved, and $\tilde{D}$ will move the token A moved. Recall that we chose a coupling of $\mathcal{G}_1$ and $\mathcal{G}_2$ that uses the same bit-strings for the corresponding tokens. Therefore, after this second move, the tokens retake identical positions in the two games.

From Case 2 we gather that $\mathcal{G}_1$ and $\mathcal{G}_2$ will reach token positions of the form $(k, k+1)$ at the same time. From Case 1, A and $\tilde{A}$ both move the same token at this point. This covers all possible positions, so the games will end at the same time, giving $M_{\tilde{A},\tilde{D}}(i,j) = M_{A,C_A}(i,j)$. ∎

Lemma 5.1.2 *If G is a lazy one-way cycle then $M^G_{\tilde{D},\tilde{A}}(i,j) \geq M^G_{D,A}(i,j)$.*

Proof. The proof is analogous to Lemma 5.1.1 using $\mathcal{G}_1 = (G, \tilde{D}, \tilde{A}, i, j)$ and $\mathcal{G}_2 = (G, D, C_D, i, j)$. ∎

Theorem 5.1.3 *If G is a lazy one-way cycle of order $n+1$ then $M^G_{A,D}(i,j) = M_{\tilde{A},\tilde{D}}(i,j)$ and $M^G_{D,A}(i,j) = M_{\tilde{D},\tilde{A}}(i,j)$. In other words, $\tilde{A}$ and $\tilde{D}$ are optimal strategies for Angel and Demon, respectively.*

Proof. First we show $M_{A,D}(i,j) = M_{\tilde{A},\tilde{D}}(i,j)$. Assuming that $i = 0$ and $j \leq \lfloor \frac{n+1}{2} \rfloor$, we have

$$\begin{aligned} M_{\tilde{A},\tilde{D}}(0,j) &\leq M_{A,D}(0,j) \\ &\leq 1 + \alpha M_{D,A}(1,j) + (1-\alpha) M_{D,A}(0,j) \\ &\leq 1 + \alpha M_{\tilde{D},\tilde{A}}(1,j) + (1-\alpha) M_{\tilde{D},\tilde{A}}(0,j) \\ &= M_{\tilde{A},\tilde{D}}(0,j), \end{aligned}$$

where the first and third inequalities hold by Lemmas 5.1.1 and 5.1.2, respectively. Each relation must hold with equality, so $M_{\tilde{A},\tilde{D}}(i,j) = M_{A,D}(i,j)$. Similarly, it follows that $M_{D,A}(i,j) = M_{\tilde{D},\tilde{A}}(i,j)$ because

$$\begin{aligned} M_{\tilde{D},\tilde{A}}(0,j) &\geq M_{D,A}(0,j) \\ &\geq 1 + \alpha M_{A,D}(0,j+1) + (1-\alpha) M_{A,D}(0,j) \\ &\geq 1 + \alpha M_{\tilde{A},\tilde{D}}(0,j+1) + (1-\alpha) M_{\tilde{A},\tilde{D}}(0,j) \\ &= M_{\tilde{D},\tilde{A}}(i,j). \end{aligned}$$

∎

The use of a corrector player is crucial in relating optimal and concrete strategies. We use this proof technique throughout the paper, as we did in this section, to prove strategies optimal.

5.2 Game Length for $\alpha = \frac{1}{2}$

In this section, we find the expected game lengths for games on a lazy one-way cycle, with $\alpha = \frac{1}{2}$. In order to gain insight into expected game length under optimal strategies, we construct a *distance digraph, G'*, as we did for the proof of Theorem 3.2.3. Let

$$V(G') = \left\{a_\ell \middle| 1 \le \ell \le \left\lfloor \frac{n+1}{2} \right\rfloor \right\} \cup \left\{d_\ell \middle| 0 \le \ell \le \left\lfloor \frac{n+1}{2} \right\rfloor \right\}.$$

Here the set $\{a_\ell\}$ ($\{d_\ell\}$) corresponds to all token distances Angel (Demon) can move on. For example, d_3 means that it is Demon's turn and the tokens are at distance 3. We note that the node d_0 corresponds to token collision on the original graph G. We let $E(G')$, be the set of weighted arcs indicating how to move on $V(G')$ to maintain the distance and player correspondence between the games. Specifically,

$$E(G') = \left\{ \begin{array}{lrl} (a_i, d_i), & 1 \le i \le \lfloor \frac{n+1}{2} \rfloor, & \text{weight } = 1-\alpha \\ (a_i, d_{i-1}), & 1 \le i \le \lfloor \frac{n+1}{2} \rfloor, & \text{weight } = \alpha \\ (d_i, a_i), & 1 \le i \le \lfloor \frac{n+1}{2} \rfloor, & \text{weight } = 1-\alpha \\ (d_i, a_{i+1}), & 1 \le i \le \lfloor \frac{n+1}{2} \rfloor - 1, & \text{weight } = \alpha \\ (d_{\lfloor \frac{n}{2} \rfloor}, a_{\lfloor \frac{n+1}{2} \rfloor - 1}), & & \text{weight } = \alpha \end{array} \right\}.$$

See Figure 5.2.

We *couple* the game on G and a random walk on G' in a natural way. If the distance between the tokens on G is ℓ and it is Demon's (Angel's) turn then place a token on d_d (a_d). This establishes an explicit correspondence between the game on G and a random walk G'. The token on G' will reach the absorbing state, d_0, at the same time that the tokens collide on G. We further simplify G' into the graph G'' by combining vertices $d_{\lfloor \frac{n+1}{2} \rfloor - 1}$ and $d_{\lfloor \frac{n+1}{2} \rfloor}$ into the single node $d'_{\lfloor \frac{n+1}{2} \rfloor - 1}$ (see Figure 5.3). A game on G is naturally coupled with the uniform random walk on G'' with absorbing state d_0. The token on G'' reaches d'_0 when the tokens on G collide. In particular, optimal play on G corresponds to a random walk on G'' with absorbing state d'_0.

Therefore the expected game length on G,

$$M_{A,D}(0, \ell) = H_{G''}(a_\ell, 0)$$

$$M_{D,A}(0, \ell) = H_{G''}(d_\ell, 0)$$

when the tokens are distance ℓ apart. This gives the following theorems.

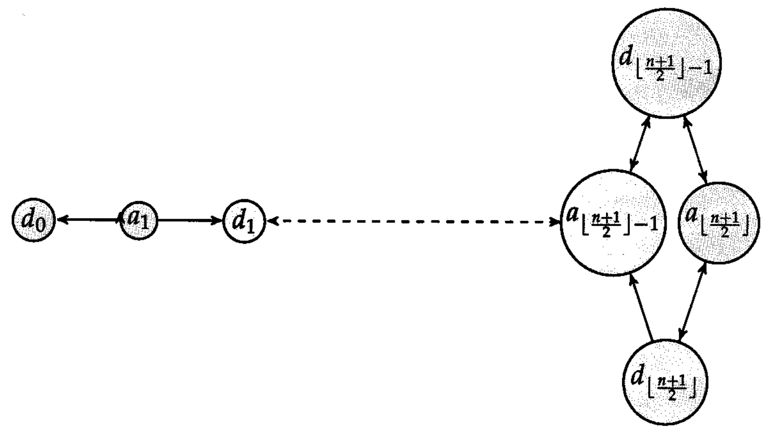

Figure 5.2: *distance digraph G'*

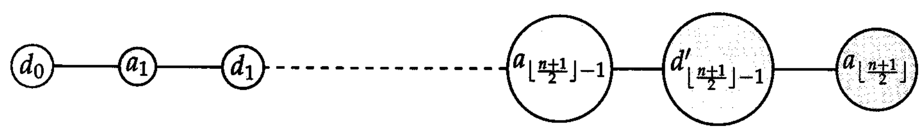

Figure 5.3: G″ path

Theorem 5.2.1 *On a lazy one-way cycle G of order $n+1$, where n is odd and $\alpha = \frac{1}{2}$,*

$$M_{A,D}(i,j) = n^2 - (n - (2 \times d_G(i,j) - 1))^2,$$

and

$$M_{D,A}(i,j) = \begin{cases} n^2 - (n - 2 \times d_G(i,j))^2 & d_G(i,j) \neq \frac{n+1}{2} \\ n^2 - (n - (2 \times d_G(i,j-2)))^2 & \textit{otherwise.} \end{cases}$$

Proof. Since $|V(G'')| = n+1$, this theorem follows directly from the well-known fact that

$$H_{P_{n+1}}(j,0) = n^2 - (n-j)^2.$$

∎

Theorem 5.2.2 *On a lazy one-way cycle of order $n+1$, where n is even and $\alpha = \frac{1}{2}$,*

$$M_{A,D}(i,j) = (n-1)^2 - ((n-1) - (2 \times d_G(i,j) - 1))^2,$$

and

$$M_{D,A}(i,j) = \begin{cases} (n-1)^2 - ((n-1) - 2 \times d_G(i,j))^2 & d_G(i,j) \neq \frac{n}{2} \\ (n-1)^2 - ((n-1) - (2 \times d_G(i,j-2)))^2 & \textit{otherwise.} \end{cases}$$

Proof. This proof is analogous to the proof of Theorem 5.2.1, with $|V(G'')| = n$. ∎

5.3 Biased Cycle

We define a biased cycle of order $n+1$ as a directed graph G with

$$\begin{aligned} V(G) &= \{i \mid i \in \{0,\ldots,n\}\}, \\ E(G) &= \{(i, i+1 \bmod n+1) \mid i \in \{0,\ldots,n\}\} \cup \{(i, i-1 \bmod n+1) \mid i \in \{0,\ldots,n\}\}, \end{aligned}$$

where

$$p_{i,j} = \begin{cases} \alpha & j = i+1 \\ 1-\alpha & j = i-1 \\ 0 & \text{otherwise.} \end{cases}$$

We assume that $\alpha \geq \frac{1}{2}$. Figure 5.4 demonstrates a biased cycle of order 6.

Theorem 5.3.1 *If G is a biased cycle of order $n+1$ then in position $(0,v)$, where $v \leq \lfloor \frac{n+1}{2} \rfloor$, an optimal strategy for Angel is to move the token at vertex 0, and an optimal strategy for Demon is to move the token at v.*

By rotational symmetry, Theorem 5.3.1 yields optimal strategies for Angel and Demon at every position. The proof of Theorem 5.3.1 is analogous to that of Theorem 5.1.3. In positions of the form $(0,v)$ we set $\tilde{A}$ to move the token at 0, and $\tilde{D}$ to move the token at v. C_A and C_D follow the correcting strategy defined in section 5.1, but with respect to the redefined $\tilde{A}$ and $\tilde{D}$. The proof of Theorem 5.3.1 requires the following 2 lemmas.

Lemma 5.3.2 *On a biased cycle G of order $n+1$, $M^G_{\tilde{A},\tilde{D}}(0,v) \leq M^G_{\tilde{A},D}(0,v)$ when $v \leq \lfloor \frac{n+1}{2} \rfloor$.*

Proof. Since

$$M_{A,C_A}(0,v) \leq M_{A,D}(0,v)$$

it suffices to show that

$$M_{\tilde{A},\tilde{D}}(0,v) = M_{A,C_A}(0,v).$$

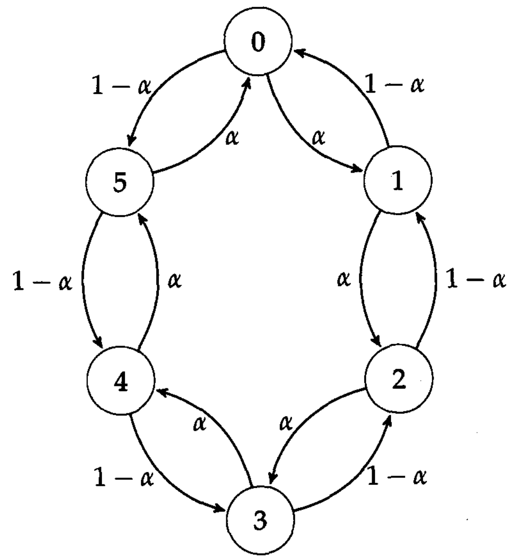

Figure 5.4: Biased Cycle of Order 6

We consider 2 cases.
Case 1: $v = 1$.
We have

$$\begin{aligned} M_{A,D}(0,1) &= 1 + \min \begin{cases} \alpha M_{D,A}(1,1) + (1-\alpha) M_{D,A}(n,1) \\ (1-\alpha) M_{D,A}(0,0) + \alpha M_{D,A}(0,2) \end{cases} \\ &= 1 + \min \begin{cases} (1-\alpha) M_{D,A}(n,1) \\ \alpha M_{D,A}(0,2) \end{cases} \\ &= 1 + (1-\alpha) M_{D,A}(n,1). \end{aligned}$$

Indeed, since $\alpha \geq \frac{1}{2}$, we have

$$(1-\alpha) M_{D,A}(n,1) \leq \alpha M_{D,A}(n,1) = \alpha M_{D,A}(0,2),$$

by rotational symmetry.

Therefore optimal Angel moves the token at 0. This is the same token chosen by the concrete strategy $\tilde{A}$ in the same configuration.
Case 2: $1 < v \leq \lfloor \frac{n+1}{2} \rfloor$.
The argument for this case and the rest of the proof is identical to the one found under Case 2 in the proof of Lemma 5.1.1. ∎

Lemma 5.3.3 *On a biased cycle G of order $n+1$, $M^G_{\tilde{D},\tilde{A}}(0,v) \leq M^G_{\tilde{D},A}(0,v)$ when $v \leq \lfloor \frac{n+1}{2} \rfloor$.*

Proof. This proof is analogous to the proof of Lemma 5.3.2. ∎

Now we can provide the proof of Theorem 5.3.1.

Proof of Theorem 5.3.1

First we show $M_{A,D}(i,j) = M_{\tilde{A},\tilde{D}}(i,j)$. We have

$$\begin{aligned} M_{\tilde{A},\tilde{D}}(0,j) &\leq M_{A,D}(0,j) \\ &\leq 1 + \alpha M_{D,A}(1,j) + (1-\alpha) M_{D,A}(n,j) \\ &\leq 1 + \alpha M_{\tilde{D},\tilde{A}}(1,j) + (1-\alpha) M_{\tilde{D},\tilde{A}}(n,j) \\ &= M_{\tilde{A},\tilde{D}}(0,j), \end{aligned}$$

where the first and third inequalities hold by Lemmas 5.3.2 and 5.3.3, respectively. Each relation must hold with equality, so $M_{\tilde{A},\tilde{D}}(i,j) = M_{A,D}(i,j)$. Similarly, it follows that $M_{D,A}(i,j) = M_{\tilde{D},\tilde{A}}(i,j)$ because

$$\begin{aligned} M_{\tilde{D},\tilde{A}}(0,j) &\geq M_{D,A}(0,j) \\ &\geq 1 + \alpha M_{A,D}(0,j+1) + (1-\alpha) M_{A,D}(0,j-1) \\ &\geq 1 + \alpha M_{\tilde{A},\tilde{D}}(0,j+1) + (1-\alpha) M_{\tilde{A},\tilde{D}}(0,j-1) \\ &= M_{\tilde{D},\tilde{A}}(i,j). \end{aligned}$$

∎

In this chapter we established a methodology for solving our game on two distinct graph structures. In the following chapter, we will use a similar methodology to find the optimal strategies on any path.

Chapter 6

Path

In Chapter 3, we characterized the game on P_3 and P_4. We now consider the game played on the path P_{n+1}. We prove that if Angel (Demon) always moves the token closest (furthest) to a leaf then he is playing optimally. We begin by introducing the necessary terminology.

6.1 Alignment Locking Pairs

Strategy Stealing

We revisit strategy stealing and coupling by considering games where two tokens are involved.

Example 6.1.1 Consider the two games $\mathcal{G}_1 = (P_{n+1}, S_1, a, b), \mathcal{G}_2(P_{n+1}, S_2, c, d)$. We have two infinite bit-strings $B_L = \{b_1, \dots\}$ and $B_R = \{b'_1, \dots\}$, where $b_i, b'_i \in \{0, 1\}$. Every time the left (right) token is moved, we use the next element in B_L (B_R) as the coin flip.

S_1 *steals* the moves of S_2 as follows. We will say that games are *normally coupled* when tokens of the same relative position (left or right) share a bit-string. At each time step, S_1 waits for S_2 to choose the token to move (left or right) and S_1 chooses the same token in his game. The tokens are then moved simultaneously using the same bit from B_L or B_R. The advantage to this coupling is that if the token positions in $\mathcal{G}_1$ and $\mathcal{G}_2$ become identical, then the games maintain identical token positions for the rest of play. In particular, if the tokens align then $\mathcal{G}_1$ and $\mathcal{G}_2$ will finish at the same time. $\diamondsuit$

Normally Aligned games

We play two games simultaneously, $\mathcal{G}_1 = (P_{n+1}, S_{11}, S_{12}, a, b)$ and $\mathcal{G}_2 = (P_{n+1}, S_{21}, S_{22}, a, b)$. Since the games have the same starting position, they are *normally aligned*. For a strategy S we use $\sigma(S)$ to denote stealing S's strategy, using a normal coupling.

Example 6.1.2 If $S_{12} = \sigma(S_{21})$ and $S_{22} = \sigma(S_{11})$ then

$$(P_{n+1}, S_{11}, \sigma(S_{21}), a, b) \equiv (P_{n+1}, \sigma(S_{11}), S_{21}, a, b) \equiv (P_{n+1}, S_{11}, S_{22}, a, b)$$

because these coupled games are always in identical states. Hence,

$$M_{S_{11},\sigma(S_{21})}(a,b) = M_{\sigma(S_{11}),S_{21}}(a,b) = M_{S_{11},S_{22}}(a,b).$$

This example plays out trivially since the games are always in identical states and using identical strategies. We consider more complicated strategy stealing configurations below. ◇

Mirror aligned games

Tokens in position (a, b) have a natural mirror symmetry with tokens in position $(n-b, n-a)$. We can use coupling and strategy stealing so that the two games, $\mathcal{G}_1 = (P_{n+1}, S_{11}, S_{12}, a, b)$ and $\mathcal{G}_2 = (P_{n+1}, S_{21}, S_{22}, n-b, n-a)$, play out identically. We allow for 2 random bit-strings, B_L and B_R. We use B_L when moving tokens starting on a in $\mathcal{G}_1$ and $n-a$ in $\mathcal{G}_2$. We use B_R when moving tokens starting on b in $\mathcal{G}_2$, $n-b$ in $\mathcal{G}_2$. In this case, the coin flips are *reverse coupled*. When a token in $\mathcal{G}_1$ moves left (right), the corresponding token in $\mathcal{G}_2$ moves right (left). We use $\sigma'(S)$ to denote stealing strategy S under a reverse coupling.

Example 6.1.3 Let $S_{11} = \sigma'(S_{21})$ and $S_{22} = \sigma'(S_{12})$. Note that $(P_{n+1}, \sigma'(S_{21}), S_{12}, a, b)$ and $(P_{n+1}, S_{21}, \sigma'(S_{12}), n-b, n-a)$ mirror each other. Hence,

$$M^{P_{n+1}}_{\sigma'(S_{21}),S_{12}}(a,b) = M^{P_{n+1}}_{S_{21},\sigma'(S_{12})}(n-b, n-a).$$

◇

Starting out of alignment

Two games are *aligned* if they are either normally or mirror aligned. In this section we study pairs of games whose starting positions are not aligned.

We consider strategies that react to the first time that the games become aligned. We note that one game may complete before alignment occurs. However, if the two games do align at some point, then our choice of strategies will guarantee that they remain aligned thereafter.

Consider the games $(P_{n+1}, S_1, L_2, a, b)$ and $(P_{n+1}, L_1, S_2, c, d)$ starting in arbitrary initial locations (a, b) and (c, d). S_1 and S_2 are some fixed strategies. We choose L_1 and L_2 (as described below) to be strategies that will lock the games in alignment if the game states become aligned. In particular, let T_a (T_{ma}) be the first time the games align (mirror align), $T = \min\{T_a, T_{ma}\}$, and let t be the current time. If one of the games finishes before aligning then we set $T_a = \infty$ and $T_{ma} = \infty$. Define *locking strategies*

$$L_1 = \begin{cases} \sigma(S_1) & : t \geq T_a, \\ \sigma'(S_1) & : t \geq T_{ma}, \\ \sigma(S_1) & : t < T, \end{cases}$$

$$L_2 = \begin{cases} \sigma(S_2) & : t \geq T_a, \\ \sigma'(S_2) & : t \geq T_{ma}, \\ \sigma(S_2) & : t < T. \end{cases}$$

These strategies have 2 phases: pre-alignment and post-alignment. If we achieve an alignment while both games are active, we are assured that they will remain aligned thereafter and finish together. We call coupled games with this property *alignment locking games*. (Note that the strategies for $t < T$ could be replaced with an arbitrary strategy, and the games maintain the alignment locking property.)

Example 6.1.4 Let S_1, S_2 be arbitrary strategies. The games $(P_5, S_1, L_2, 0, 3)$ and $(P_5, L_1, S_2, 1, 4)$ start in mirror alignment ($T_{ma} = 0$). In other words, they start in phase 2 (post-alignment). With the reverse coupling in place, the coupled games proceed under the strategies above as shown in Figure 6.1.

The games remain aligned and finish at the same time. ♢

Example 6.1.5 Let S_1, S_2 be arbitrary strategies. Consider the play of the games $(P_6, S_1, L_2, 0, 4)$ and $(P_6, L_1, S_2, 1, 3)$, shown in Figure 6.2. We have $T_{ma} = 3$, $T_a = 4$, and so the games are not aligned at $t = 1$. Hence, at $t = 3$, opposite tokens are chosen. Pre-alignment, both games proceed under arbitrarily chosen strategies and distinct bit-string.

♢

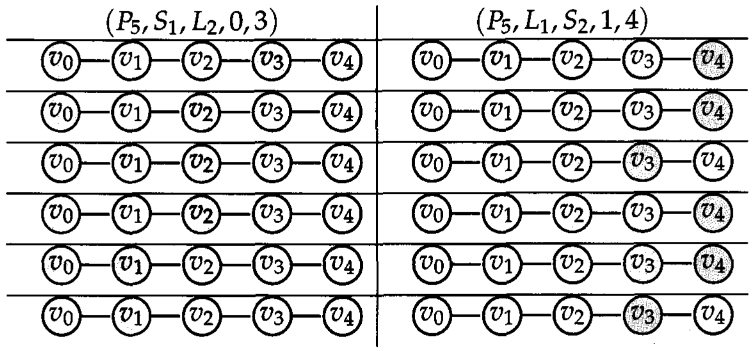

Figure 6.1: $(P_5, S_1, L_2, 0, 3)$ and $(P_5, L_1, S_2, 1, 4)$

6.2 Optimal Strategies on the Path

We use alignment locking games to identify optimal strategies for Angel and Demon. We employ corrector players as in the proof of Theorem 5.1.3 and Theorem 5.3.1. Recall that A, D are the optimal Angel, Demon strategies, respectively. Consider the games (P_{n+1}, D, A^*, a, b), (P_{n+1}, D^*, A, c, d) where A^*, D^* are the locking strategies

$$A^* = \begin{cases} \sigma(A) & : t \geq T_a \\ \sigma'(A) & : t \geq T_{ma} \\ \sigma(A) & : t \leq T, \end{cases}$$

and

$$D^* = \begin{cases} \sigma(D) & : t \geq T_a \\ \sigma'(D) & : t \geq T_{ma} \\ \sigma(D) & : t \leq T. \end{cases}$$

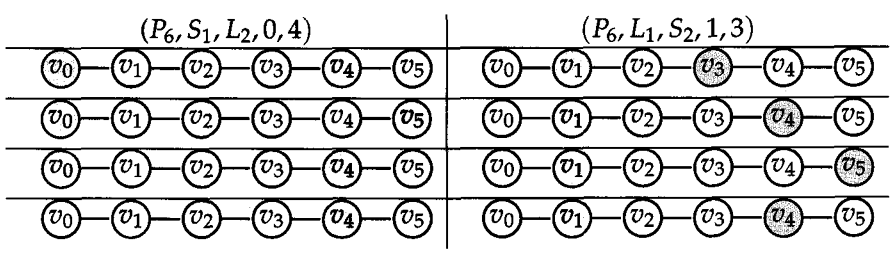

Figure 6.2: Trajectory of $(P_6, S_1, L_2, 0, 4)$ and $(P_6, L_1, S_2, 1, 3)$

Before alignment, A^* steals A's strategy using a normal coupling. If the games align, then A^* reacts to the type of alignment. D^* behaves analogously. With these strategies, (P_{n+1}, D, A^*, a, b) and (P_{n+1}, D^*, A, c, d), are an *alignment locking pair*.

First we identify the behavior of the optimal strategies for adjacent tokens. This case (which must occur directly before a collision) will be crucial to characterizing the strategies for the other positions.

Lemma 6.2.1 *If* $a+1 \leq \lceil \frac{n}{2} \rceil$ *then* $M_{A,D}^{P_{n+1}}(a, a+1) = 1 + \frac{1}{2} M_{D,A}^{P_{n+1}}(a-1, a+1)$.

This lemma states that if the tokens are adjacent, then moving the token with the minimum distance from a leaf is optimal for Angel.

Proof. We have

$$\begin{aligned} M_{A,D}(a, a+1) &= 1 + \frac{1}{2} \min \left\{ \begin{array}{l} M_{D,A}(a-1, a+1) \\ M_{D,A}(a, a+2), \end{array} \right. \\ &= 1 + \frac{1}{2} \min \left\{ \begin{array}{l} M_{D,A}(a-1, a+1) \\ M_{D,A}(a, a+2) \end{array} \right. . \end{aligned}$$

In addition,

$$M_{D,A}(a-1, a+1) = M_{D,A^*}(a-1, a+1), \tag{6.1}$$

and

$$M_{D^*,A}(a, a+2) = M_{D,A}(a, a+2) \tag{6.2}$$

because A^* (D^*) cannot be better than the optimal strategy A (D).

The case $a = 0$, follows directly from Lemma 3.3.2, and is clear since it ends in one step.

Now suppose $0 < a \leq \lceil \frac{n}{2} \rceil - 1$. By 6.1 and 6.2, it suffices to show that

$$M_{D,A^*}(a-1, a+1) \leq M_{D^*,A}(a, a+2).$$

The games $\mathcal{G}_1 = (P_{n+1}, D, A^*, a-1, a+1)$ and $\mathcal{G}_2 = (P_{n+1}, D^*, A, a, a+2)$ are an alignment locking pair. If the games become aligned then they meet at exactly the same time. So we need only consider the cases where no game alignment occurs before one of the games completes.

A token on a leaf can bounce. A *bounce* occurs when a token on a leaf is chosen, and the bit corresponding to it indicates one should move off the path. For example, if a token on the left leaf of path is chosen, and the corresponding bit indicates a step left, then a bounce occurs: the token "bounces" one step right. We will see that bounces are advantageous to Angel.

Suppose no alignment occurs before the tokens meet. Let β be the total number of bounces in both games before either game ends. Let τ_L and τ_L' (τ_R and τ_R') be the left-hand side (right-hand side) tokens on $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively. Recall that we are assuming without loss of generality that, in starting positions, τ_L and τ_L' are at least as close to leaves as the right-hand side tokens, τ_R and τ_R'.

Case 1: β is even.

If no bounce occurs then the expected collision times of both games are equal since $|a-b| = |c-d|$ (Theorem 3.2.3). If the first bounce occurs with τ_R', then token positions (k,m) and $(n-m,n-k)$ must occur prior to the bounce. In other words, we must encounter a mirror symmetry (contradicting our assumption), which aligns the games thereafter. Therefore, the first bounce on each path must occur with τ_L.

We claim that the token distances in $\mathcal{G}_1$ and $\mathcal{G}_2$ are equal when β is even. Immediately after the first bounce, the game states are $(1,k)$ in $\mathcal{G}_1$ and $(0,k+1)$ on $\mathcal{G}_2$, where $k \in (2,n-1)$. We have not yet attained a symmetry lock, so τ_L, τ_L' are still normally coupled and τ_R, τ_R' are still normally coupled. In addition, the locations of tokens in the first game are interior to the locations of the tokens in the second game. Therefore, a token on $\mathcal{G}_2$ must bounce next.

If τ_L' bounces first, this yields token positions $(0,m)$ on $\mathcal{G}_1$ and $(1,m+1)$ on $\mathcal{G}_2$, where $m \in [2,n-1]$. The tokens are once again in positions that occur when no bounce occurs and reestablish an equal distance between tokens.

If τ_R' token bounces first, we simply reverse the indexing of the path, and the argument in the previous paragraph holds, again re-establishing an equal distance between tokens. In other words, on every second bounce, the configuration is consistent with a game pair in which no bounces have occurred.

Case 2: β is odd.

As established above, when β becomes odd, the tokens are in positions $(1,k)$ in $\mathcal{G}_1$ and $(0,k+1)$ in $\mathcal{G}_2$ and they are normally coupled. Therefore, if the game ends after an odd number of bounces, and no alignment occurs, then $\mathcal{G}_1$ will finish faster than $\mathcal{G}_2$, since the tokens in $\mathcal{G}_1$ are closer.

In each case $\mathcal{G}_1$ finishes at least as fast as $\mathcal{G}_2$. Therefore,

$$M_{D,A^*}(a-1,a+1) \leq M_{D^*,A}(a,a+2), \text{when } 0 < a \leq \left\lceil \frac{n}{2} \right\rceil - 1,$$

so $M_{D,A}(i-1,i+1) \leq M_{D,A}(i,i+2)$ for all i. Hence, $M_{A,D}(a,a+1) = 1 + M_{D,A}(\overline{a},a+1)$. ∎

Lemma 6.2.2 *If* $a+1 \le \lceil \frac{n}{2} \rceil$ *then* $M^{P_{n+1}}_{D,A}(a,a+1) = 1 + \frac{1}{2} M^{P_{n+1}}_{A,D}(a,a+2)$.

Proof. The proof is analogous to that of Lemma 6.2.1 with $\mathcal{G}_1 = (P_{m+1}, A^*, D, a-1, a+1)$ and $\mathcal{G}_2 = (P_{n+1}, D^*, A, a, a+2)$. ▮

We let $\tilde{A}$ ($\tilde{D}$) be the angel (demon) player that always moves the token closest (furthest) to (from) an leaf. We prove that these concrete strategies are optimal. The crucial work occurs in the next 2 lemmas.

Lemma 6.2.3 $M_{A,D}(i,j) \ge M_{\tilde{A},\tilde{D}}(i,j)$

Proof. This proof is similar to that of Lemma 5.1.1. Let C_A be the angel *corrector player*. C_A plays like $\tilde{D}$ until A does not move like $\tilde{A}$, in which case C_A responds by making the move that $\tilde{A}$ would have moved. By Lemma 6.2.1, we know A moves like $\tilde{A}$ in token position $(i, i+1)$. Furthermore, $M_{A,D}(i,j) \ge M_{A,C_A}(i,j)$. So we prove $M_{A,C_A}(i,j) \ge M_{\tilde{A},\tilde{D}}(i,j)$.

Consider the first instance, with token positions (i,j), where $\mathcal{G}_1 = (P_{n+1}, A, C_A, i, j)$ deviates from $\mathcal{G}_2 = (P_{n+1}, \tilde{A}, \tilde{D}, i, j)$. By Lemma 6.2.1 and Lemma 6.2.2, $j \ge i+2$. Recall that we are assuming $i \le n-j$ and so A must have moved τ_R. There are 2 cases to consider. Assume we have moved τ_L a total of $r-1$ times and τ_R a total of $s-1$ times.
Let $B_L(r)$ ($B_R(s)$) denote the r^{th} (s^{th}) bit of B_L (B_R).
Case 1: $B_R(s) = 0$
The resulting token position in $\mathcal{G}_1$ is $(i, j-1)$. Because $\mathrm{d}(i,j) \ge 2$ the game continues. On the following move C_A will move τ_L. The resulting position is $(i+1, j-1)$ or $(i-1, j-1)$ if $B_L(r) = 0$ or $B_L(r) = 1$, respectively.
Since we have predetermined bits for each token it must hold that after these two moves, the token position for $\mathcal{G}_2$ is $(i+1, j-1)$ or $(i-1, j-1)$ if $B_L(r) = 0$ or $B_L(r) = 1$, respectively. In other words, the games realign after 2 moves.
Case 2: $B_R(s) = 1$.
Again the games realign after 2 moves. One move after the deviation, the resulting token position for $\mathcal{G}_1$ ($\mathcal{G}_2$) is $(i+1, j+1)$ or $(i-1, j+1)$ if $B_L(r) = 0$ or $B_L(r) = 1$, respectively. We argue similar to Case 1 to show realignment.

Hence, we may deduce that any deviation of $\mathcal{G}_1$ from $\mathcal{G}_2$ results in the tokens realigning on the very next move, prior to the next deviation.

To complete the proof, we must take into account the parity of $\mathrm{d}(i,j)$.

If $d(i,j)$ is odd then both games will always reach token position $(k, k+1)$ at the same time. By Lemma 6.2.1, we know that in these positions they will move the same token. Therefore, it must hold that in this case $M_{A,C_A}(i,j) = M_{\tilde{A},\tilde{D}}(i,j)$.

If $d(i,j)$ is even then tokens in the game maintain an even distance. Thus, in the step after a deviation, both games will either end or realign and continue. So in this case too, $M_{A,C_A}(i,j) = M_{\tilde{A},\tilde{D}}(i,j)$. We conclude that $M_{A,D}(i,j) \geq M_{\tilde{A},\tilde{D}}(i,j)$. ∎

Lemma 6.2.4 $M_{D,A}(i,j) \leq M_{\tilde{D},\tilde{A}}(i,j)$

Proof. The proof is analogous to that of Lemma 6.2.3 with $\mathcal{G}_1 = (P_{n+1}, D, C_D, i, j)$ and $\mathcal{G}_1 = (P_{n+1}, \tilde{D}, \tilde{A}, i, j)$ ∎

We now prove our main results.

Theorem 6.2.5 $M^{P_{n+1}}_{\tilde{A},\tilde{D}}(i,j) = M^{P_{n+1}}_{A,D}(i,j)$.

Proof. We first show that $M^{P_{n+1}}_{\tilde{A},\tilde{D}}(i,j) = M^{P_{n+1}}_{A,D}(i,j)$. We must consider 3 cases.
Case 1: $j = i + 1$.
If $i = 0$, then $M_{A,D}(0,1) = 1 = M_{\tilde{A},\tilde{D}}(0,1)$ by Lemmas 6.2.1 and 6.2.2. If $i > 0$, it follows from Lemma 6.2.1, Lemma 6.2.3 and Lemma 6.2.4 that

$$\begin{aligned} M_{\tilde{A},\tilde{D}}(i,i+1) &\leq M_{A,D}(i,i+1) \\ &= 1 + \frac{1}{2} M_{D,A}(i-1,i+1) \\ &\leq 1 + \frac{1}{2} M_{\tilde{D},\tilde{A}}(i-1,i+1) \\ &= M_{\tilde{A},\tilde{D}}(i,i+1). \end{aligned}$$

So we have equality throughout.
Case 2: $i = 0$ and $j \geq i + 2$.
We have a similar sandwiching of inequalities. Using Lemma 6.2.3 and Lemma 6.2.4

$$\begin{aligned} M_{\tilde{A},\tilde{D}}(0,j) &\leq M_{A,D}(0,j) \\ &\leq 1 + M_{D,A}(1,j) \\ &\leq 1 + M_{\tilde{D},\tilde{A}}(1,j) \\ &= M_{\tilde{A},\tilde{D}}(0,j). \end{aligned}$$

Case 3: $i > 0$ and $j \geq i+2$.
Employing Lemma 6.2.3 and Lemma 6.2.4 yields

$$\begin{aligned} M_{\tilde{A},\tilde{D}}(i,j) &\leq M_{A,D}(i,j) \\ &\leq 1+\frac{1}{2}\left(M_{D,A}(i+1,j)+M_{D,A}(i-1,j)\right) \\ &\leq 1+\frac{1}{2}\left(M_{\tilde{D},\tilde{A}}(i+1,j)+M_{\tilde{D},\tilde{A}}(i-1,j)\right) \\ &= M_{\tilde{A},\tilde{D}}(i,j). \end{aligned}$$

∎

Theorem 6.2.6 $M^{P_{n+1}}_{\tilde{D},\tilde{A}}(i,j) = M^{P_{n+1}}_{D,A}(i,j)$

Proof. As in the previous proof, we must consider 3 cases.
Case 1: $j = i+1$.
We have

$$\begin{aligned} M_{\tilde{D},\tilde{A}}(i,i+1) &\geq M_{D,A}(i,i+1) \\ &= 1+\frac{1}{2}M_{A,D}(i,i+2) \\ &\geq 1+\frac{1}{2}M_{\tilde{A},\tilde{D}}(i,i+2) \\ &= M_{\tilde{D},\tilde{A}}(i,i+1), \end{aligned}$$

where the first inequality follows from 6.2.1 and the third inequality is due to Lemma 6.2.3.
Case 2: $i = 0$ and $j \geq i+2$.

$$\begin{aligned} M_{\tilde{D},\tilde{A}}(0,j) &\geq M_{D,A}(0,j) \\ &\geq 1+M_{A,D}(1,j) \\ &\geq 1+M_{\tilde{A},\tilde{D}}(1,j) \\ &= M_{\tilde{D},\tilde{A}}(0,j). \end{aligned}$$

Case 3: $i > 0$ and $j \geq i+2$.
We have

$$\begin{aligned} M_{\tilde{D},\tilde{A}}(i,j) &\geq M_{D,A}(i,j) \\ &\geq 1+\frac{1}{2}\left(M_{A,D}(i,j+1)+M_{A,D}(i,j-1)\right) \\ &\geq 1+\frac{1}{2}\left(M_{\tilde{A},\tilde{D}}(i,j+1)+M_{\tilde{A},\tilde{D}}(i,j-1)\right) \\ &= M_{\tilde{D},\tilde{A}}(i,j), \end{aligned}$$

where the first inequality follows from Lemma 6.2.4 and the third inequality follows from Lemma 6.2.3. ∎

We have shown that, when a choice exists, an optimal strategy for Angel is to always move the token closest to a leaf, and an optimal strategy for Demon is to always move the token farthest from a leaf.

6.3 Implications for Meeting Times

With pure strategies for Angel and Demon established we can begin to derive expected game length relations. In particular, we can construct a graph G, on which we will place a *distance token*, as we did in the proof of Theorem 3.2.3 and Section 5.2.

Consider (P_{n+1}, A, D, i, j), where $V(P_{n+1}) = \{0, \ldots, n\}$ and $|i-j|$ is odd. We let

$$\begin{aligned} V(G) = \{&a_{(i,j)} \mid \forall\, i,j \in V(P_{n+1}) \text{ where } |i-j| \text{ is odd}\} \cup \\ &\{d_{(i,j)} \mid \forall\, i,j \in V(P_{n+1}) \text{ where } |i-j| \text{ is even}\}. \end{aligned}$$

As a pair of tokens moves within P_{n+1}, a single distance token moves around G according the following rule: if the tokens on P_{n+1} are at vertices i and j, with $|i-j|$ odd, then the location $a_{k'}(d_{k'})$ of the distance token is given by $k' = |i-j|$. We let $\mathrm{E}(G)$ be the set of arcs with the smallest cardinality allowing the token to move along the vertices according to the rule. The distance token takes a uniform random walk on G, which has the same expected game length as (P_{n+1}, A, D, i, j). We similarly construct G for (P_{n+1}, D, A, i, j) and $|i-j|$ even.

We use the random walk of the distance token in order to find the expected game lengths.

Example 6.3.1 Let

$$(M_A^{P_5})_{i,j} = \left(\begin{array}{ll} M_{A,D}^{P_5}(i,j), & \text{if } |i-j| \text{ is odd;} \\ M_{D,A}^{P_5}(i,j), & \text{if } |i-j| \text{ is even} \end{array} \right),$$

$$(M_D^{P_5})_{i,j} = \left(\begin{array}{ll} M_{D,A}^{P_5}(i,j), & \text{if } |i-j| \text{ is odd;} \\ M_{A,D}^{P_5}(i,j), & \text{if } |i-j| \text{ is even} \end{array} \right).$$

In particular, $(M_A^{P_5})_{i,j}$ is the expected game length on P_5 when Angel moves last and the initial token position is (i,j). $(M_D^{P_5})_{i,j}$ is defined similarly. We calculate $M_A^{P_5} = \{(M_A^{P_5})_{i,j} \mid i,j \in \{0,1,2,3,4\}\}$ and $M_D^{P_5} = \{(M_D^{P_5})_{i,j} \mid i,j \in \{0,1,2,3,4\}\}$ through a system of linear equations.

$$M_A = \begin{pmatrix} 0 & 1 & \frac{16}{3} & \frac{23}{3} & \frac{26}{3} \\ 1 & 0 & \frac{11}{3} & \frac{20}{3} & \frac{23}{3} \\ \frac{16}{3} & \frac{11}{3} & 0 & \frac{11}{3} & \frac{16}{3} \\ \frac{23}{3} & \frac{20}{3} & \frac{11}{3} & 0 & 1 \\ \frac{26}{3} & \frac{23}{3} & \frac{16}{3} & 1 & 0 \end{pmatrix}, \quad M_D = \begin{pmatrix} 0 & 4 & 6 & 9 & 10 \\ 4 & 0 & 5 & 8 & 9 \\ 6 & 5 & 0 & 5 & 6 \\ 9 & 8 & 5 & 0 & 4 \\ 10 & 9 & 6 & 4 & 0 \end{pmatrix}$$

◇

We encounter some interesting patterns when comparing $M_A^{P_{n+1}}$ and $M_D^{P_{n+1}}$ (or equivalently $M_{A,D}^{P_{n+1}}(i,j)$ and $M_{D,A}^{P_{n+1}}(i,j)$), for odd n and even $|i-j|$.

Example 6.3.2 Let $n = 7$. We define a matrix $\Delta = \{\Delta_{i,j} \mid 0 \leq i,j \leq n\}$, where $\Delta_{i,j} = (M_D^{P_{n+1}})_{i,j} - (M_A^{P_{n+1}})_{i,j}$.

$$M_A^{P_8} = \begin{pmatrix} 0 & 1 & \frac{244}{29} & \frac{401}{29} & \frac{584}{29} & \frac{709}{29} & \frac{796}{29} & \frac{825}{29} \\ 1 & 0 & \frac{151}{29} & \frac{372}{29} & \frac{535}{29} & \frac{680}{29} & \frac{767}{29} & \frac{796}{29} \\ \frac{244}{29} & \frac{151}{29} & 0 & \frac{215}{29} & \frac{428}{29} & \frac{583}{29} & \frac{680}{29} & \frac{709}{29} \\ \frac{401}{29} & \frac{372}{29} & \frac{215}{29} & 0 & \frac{243}{29} & \frac{428}{29} & \frac{535}{29} & \frac{584}{29} \\ \frac{584}{29} & \frac{535}{29} & \frac{428}{29} & \frac{243}{29} & 0 & \frac{215}{29} & \frac{372}{29} & \frac{401}{29} \\ \frac{709}{29} & \frac{680}{29} & \frac{583}{29} & \frac{428}{29} & \frac{215}{29} & 0 & \frac{151}{29} & \frac{244}{29} \\ \frac{796}{29} & \frac{767}{29} & \frac{680}{29} & \frac{535}{29} & \frac{372}{29} & \frac{151}{29} & 0 & 1 \\ \frac{825}{29} & \frac{796}{29} & \frac{709}{29} & \frac{584}{29} & \frac{401}{29} & \frac{244}{29} & 1 & 0 \end{pmatrix}$$

$$M_D^{P_8} = \begin{pmatrix} 0 & \frac{151}{29} & \frac{244}{29} & \frac{443}{29} & \frac{584}{29} & \frac{719}{29} & \frac{796}{29} & \frac{825}{29} \\ \frac{151}{29} & 0 & \frac{215}{29} & \frac{372}{29} & \frac{555}{29} & \frac{680}{29} & \frac{767}{29} & \frac{796}{29} \\ \frac{244}{29} & \frac{215}{29} & 0 & \frac{243}{29} & \frac{428}{29} & \frac{583}{29} & \frac{680}{29} & \frac{719}{29} \\ \frac{443}{29} & \frac{372}{29} & \frac{243}{29} & 0 & \frac{243}{29} & \frac{428}{29} & \frac{555}{29} & \frac{584}{29} \\ \frac{584}{29} & \frac{555}{29} & \frac{428}{29} & \frac{243}{29} & 0 & \frac{243}{29} & \frac{372}{29} & \frac{443}{29} \\ \frac{719}{29} & \frac{680}{29} & \frac{583}{29} & \frac{428}{29} & \frac{243}{29} & 0 & \frac{215}{29} & \frac{244}{29} \\ \frac{796}{29} & \frac{767}{29} & \frac{680}{29} & \frac{555}{29} & \frac{372}{29} & \frac{215}{29} & 0 & \frac{151}{29} \\ \frac{825}{29} & \frac{796}{29} & \frac{719}{29} & \frac{584}{29} & \frac{443}{29} & \frac{244}{29} & \frac{151}{29} & 0 \end{pmatrix}$$

$$\Delta = \begin{pmatrix} 0 & \frac{122}{29} & 0 & \frac{42}{29} & 0 & \frac{10}{29} & 0 & 0 \\ \frac{122}{29} & 0 & \frac{64}{29} & 0 & \frac{20}{29} & 0 & 0 & 0 \\ 0 & \frac{64}{29} & 0 & \frac{28}{29} & 0 & 0 & 0 & \frac{10}{29} \\ \frac{42}{29} & 0 & \frac{28}{29} & 0 & 0 & 0 & \frac{20}{29} & 0 \\ 0 & \frac{20}{29} & 0 & 0 & 0 & \frac{28}{29} & 0 & \frac{42}{29} \\ \frac{10}{29} & 0 & 0 & 0 & \frac{28}{29} & 0 & \frac{64}{29} & 0 \\ 0 & 0 & 0 & -\frac{20}{29} & 0 & \frac{64}{29} & 0 & \frac{122}{29} \\ 0 & 0 & \frac{10}{29} & 0 & \frac{42}{29} & 0 & \frac{122}{29} & 0 \end{pmatrix}$$

Note that for all even $|i-j|$,

$$(M_A^{P_8})_{i,j} - (M_D^{P_8})_{i,j} = M_{A,D}^{P_8}(i,j) - M_{D,A}^{P_8}(i,j) = 0.$$

In other words, when $|i-j|$ is even, then

$$M_{A,D}^{P_{n+1}}(i,j) = M_{D,A}^{P_{n+1}}(i,j).$$

$\diamondsuit$

As we continue to examine these types of matrices, for odd n and even $|i-j|$, we find that the pattern of $M_{A,D}^{P_{n+1}}(i,j) - M_{D,A}^{P_{n+1}}(i,j) = 0$ continues.

Lemma 6.3.3 *Suppose n is odd. For all even $|i-j|$, $M_{A,D}^{P_{n+1}}(i,j) = M_{D,A}^{P_{n+1}}(i,j)$.*

Proof. We normally couple $\mathcal{G}_1 = (P_{n+1}, A, D, i, j)$ with $\mathcal{G}_2 = (P_{n+1}, D, A, i, j)$. In other words, we assign identical random bit-strings to the tokens on i and on j.

Note that if $i = n-j$, $|i-j|$ is odd, and so the tokens do not begin in this position. The token closest to a leaf does not change unless we enter a symmetry position after our initial move. If there is a symmetry we may determine which token is moved; we choose to move the token not moved in the previous time step.

In the first two moves, opposite tokens are moved by Angel and Demon in $\mathcal{G}_1$ and $\mathcal{G}_2$. Therefore the tokens realign after the first two moves. Hence, the games continue to realign in every even round. In other words, if the token distance in $\mathcal{G}_1$ is even, then the token positions are identical in $\mathcal{G}_1$ and $\mathcal{G}_2$. This includes times when the tokens meet, which must occur in an even round. ▮

Corollary 6.3.4 *If n is odd and $i = n-j$ then $M_{A,D}^{P_{n+1}}(i, n-i) = M_{D,A}^{P_{n+1}}(i, n-i)$.*

Proof. Note that $(n-i) - i = n - 2i$ is odd. We have

$$M_{A,D}(i, n-i) = 1 + M_{D,A}(\bar{\imath}, n-i) = 1 + M_{A,D}(i, \overline{n-i}) = M_{D,A}(i, n-i),$$

where the second equality follows from Lemma 6.3.3. The potential token positions are the same regardless of which token is chosen. ▮

Chapter 7

Open Problems

In this paper we found optimal strategies for Angel and Demon on a variety of graphs. Our research began with the path, in hopes of finding optimal strategies for Angel and Demon on a variety of tree structures. We conjecture that an optimal strategy for Angel and Demon on any tree is analogous to the optimal strategy we found for Angel and Demon on the path. Angel should always move the token closest to a leaf, and Demon should always move the token farthest from a leaf. Below we provide a list of open problems we obtained from Coppersmith et al. (1993b), Tetali and Winkler (1993), and our own research that the interested reader can pursue.

- In the two player game, what are optimal strategies for Angel and Demon on m-ary trees?
- In the two player game, what are optimal strategies for Angel and Demon on any tree?
- In the two player game, what are optimal strategies for Angel and Demon on any graph?
- Find more accurate bounds for game length on trees of order n.
- In the single player game, does there exist a graph such that a clairvoyant Demon can force the game to never end with probability > 0?
- In the single player game, what are optimal strategies for Angel and Demon on any graph?

Bibliography

D. Aldous. Meeting times for independent Markov chains. *Stochastic Processes and Applications*, 38:185–193, 1991.

G. Brightwell and P. Winkler. Maximum hitting time for random walks on graphs. In *Proc. 22nd Annual ACM Symposium on Theory of Computing*, pages 369–378, 1990.

D. Coppersmith, P. Doyle, P. Reghavan, and M. Snir. Random walks on weighted graphs, and applications to on-line algorithms. *Journal of the Association for Computing Machinery*, 40(3):421–453, 1993a.

D. Coppersmith, P. Tetali, and P. Winkler. Collisions among random walks on a graph. *SIAM Journal on Discrete Mathematics*, 6(3):363–374, 1993b.

P. G. Doyle and J. L. Snell. *Random Walks and Electric Networks*. Mathematical Association of America, 1984.

O. Häggström. *Finite Markov Chains and Algorithmic Applications*. Cambridge University Press, 2002.

A. Israeli and M. Jalfon. Token management schemes and random walks yield self stabilizing mutual exclusion. In *Proc. 9th Annual ACM Symposium on Principles of Distributed Computing*, pages 119–131, 1990.

J. Kemeny and J. L. Snell. *Finite Markov Chains: With a New Appendix "Generalization of a Fundamental Matrix"*. Springer, 2nd edition, 1976.

R. J. Larson and M. L. Marx. *An Introduction to Mathematical Statistics and Its Applications*. Prentice Hall, 4th edition, 2005.

P. Tetali and P. Winkler. Simultaneous reversible Markov chains. In *Combinatorics, Paul Erdős is Eighty*, volume 1, pages 433–452, 1993.

D. West. *Introduction to Graph Theory*. Prentice Hall, 2nd edition, 2000.