A Thesis Submitted for the Degree of PhD at the University of Warwick

Three Essays on Sequential Learning in Search - Pursuit Games and Jury Voting Problems

by

Viciano Lee

Thesis

Submitted to the University of Warwick

in partial fulfilment of the requirements

for the admission to the degree of

Doctor of Philosophy

Warwick Business School

January 2021

THE UNIVERSITY OF WARWICK

Contents

List of Tables

List of Figures

Acknowledgments

First and foremost I would like to express my deepest appreciation to my academic supervisors, Prof. Steve Alpern and Prof. Bo Chen for providing invaluable supervision of this thesis. Their comments and suggestions have been consistently provocative and insightful and it has been an honor to be one of their Ph.D. students

I would also like to thank Prof. Jürgen Branke, Prof. Leroy White, Dr. Nalân Gülpınar, Dr. Xuan Vinh Doan, and Dr. Nursen Aydin for their valuable advice during the Ph.D. upgrade, annual reviews, and final year completion review panel.

I must also acknowledge the importance of the work of Shmuel Gal, and Jerome Casas which played a large part in drawing me to the area of search and pursuit games (Chapter 2, and Chapter 3). Special appreciation must also go to Steve Alpern and Bo Chen for their seminal work which inspired the idea for Chapter 4.

I would also like to offer thanks to the staff of the Operations group and the Doctoral Programme Office (DPO) of the Warwick Business School as well as my fellow postgraduate colleagues for creating a supportive environment which has aided me through the course of my Ph.D study.

Last but not least, I would like to express a heartfelt thank you to my family and best friends for always believing in me and encouraging me to follow my dreams.

Declarations

I declare that any material contained in this thesis has not been submitted for a degree to any other university. I further declare that a paper titled "A Stochastic Game Model of Searching Predator and Hiding Prey", drawn from Chapter 2 of this thesis, is a product of joint work with Shmuel Gal, Jerome Casas, and Steve Alpern. My main contribution is on the Learning model of the Stochastic Game. I am also involved in the discussion with Professor Alpern regarding other Sections of this paper. It has been published in *The Journal of Royal Society Interface* [12]. Furthermore, the paper "A Normal Form Game Model of Search and Pursuit", drawn from Chapter 3 of this thesis, was co-authored with Steve Alpern and I contributed 80% of this work. It is currently *in press* for the forthcoming edition of the *Advances in Dynamic Games and Applications* [14]. Finally, the paper titled "Optimizing the Ordering of Sequential Jury Voting: A Sealed Card Model", drawn from Chapter 4, is a joint project with Steve Alpern, Bo Chen, and Pan Chenxin. It is in preparation for submission [5]. All of the Propositions in this paper are done by exhaustive computation over a finite number of cases and I did 100% of the calculation in *Mathematica* (Pan also did most of the calculation in Matlab and we check the accuracy of our result by comparing the findings using these two programming tools).

I declare that my thesis consists of 26717 words.

Viciano Lee

November 2020

Abstract

This thesis consists of three papers on different topics on the sequential learning problem. The first paper is a stochastic game model of predator-prey interaction that combines search and pursuit in a single game. In this paper I introduce a novel learning model in which certain parameters of the model could in theory be learned sequentially over time. The second paper in this thesis focuses on the deterministic version of the similar search and pursuit problem. By relaxing the assumption made since Gal and Casas [36], I introduce three sub-problems to the original model and provide general propositions to each of them. The third paper departs from the area of search games and analyzes the optimal voting ordering in sequential juries. Together with Steve Alpern, Bo Chen, and Chenxin Pan, we studied a new version of the Alpern–Chen model [2]. Our paper makes the notion of "ability" more specific by introducing the sealed card model in which the state of Nature (equivalent to innocent or guilty for a legal jury) is the color of a sealed card and the jurors sample other cards in the deck as their 'signals' and they vote sequentially. I show that under this model, by voting in Alpern-Chen ordering (median ability, high ability, low ability), the jury ensures the highest average reliability and optimality fraction.

Chapter 1

Introduction

This thesis follows the Warwick Business School structure as a three-paper thesis. It is based on three journal articles, where two of them are already published and the third one is in preparation for submission. The three articles have a common theme where sequential learning is used to predict some unknown state of Nature. In the first two articles, the state of Nature could be the hiding location of a prey animal; in the third article, it could be for instance, the innocence or guilt of a defendant to be decided by a jury.

The thesis explores different aspects of sequential learning problems. The three main chapters, though distinct in their focus, are aimed at analyzing the role of sequential learning in two areas: Search and Pursuit Games, and Jury Voting Problem. Chapter 2 studies the search and pursuit interaction between a predator and a prey using stochastic games, while Chapter 3 solves similar problem to that above, using the normal form game. Chapter 4 is within the realm of a Jury Voting Problem. More specifically, this is a discrete model, expressed in an easy-to-understand setting of playing cards, which simplifies the seminal model of Alpern and Chen [2], on sequential juries. Chapter 2 and Chapter 3 are within the framework of non–cooperative zero–sum game, specifically search games, while the main research question in chapter 4 is an optimization problem solved using integer and dynamic programming. The common thread of the three chapters is to learn the state of Nature (hiding locations or correct alternative) through sequential tests, which could be searching or sampling.

The thesis is structured as follows: Chapter 2, 3, and 4 are presented in the form of papers. The first two are published and the third paper is ready for submission. Thus, each chapter consists of its own introduction, literature review, methodology, findings, and conclusion. Chapter 2 corresponds to a published paper in [12]. Chapter 3 is drawn from [14] which has been published in the most recent edition of *Advances in Dynamic Games*. Chapter 4 is an unpublished paper written with Steve Alpern, Bo Chen, and Chenxin Pan. This paper is in its final draft and is in preparation for submission [5]. Lastly, Chapter 5 summarizes the contributions of this thesis and describes future research directions. The rest of this Section summarizes each paper, giving the reader the main idea and the contributions to the literature. Section 1.4 in particular discusses the role of Operations Research in search and pursuit games and jury voting problem.

1.1 A Stochastic Game Model of Searching Predator and Hiding Prey (Chapter 2)

When the spatial density of both prey and predators is very low, the problem they face may be modelled as a two-person game (called a 'search game') between one member of each type. Following recent models of search and pursuit, we assume the prey has a fixed number of heterogeneous 'hiding' places (for example, ice holes for a seal to breathe) and that the predator (maybe polar bear) has the time or energy to search a fixed number of these. If he searches the actual hiding location and also successfully pursues the prey there, he wins the game. If he fails to find the prey, he loses. In this chapter, we modify the outcome in the case that he finds but does not catch the prey. The prey is now vulnerable to capture while relocating with risk depending on the intervening terrain. This generalizes the original games to a stochastic game framework, a first for search and pursuit games. We outline a general solution and also compute particular solutions. This modified model now has implications for the question of when to stay or leave the lair and by what routes. In particular, we find the counterintuitive result that in some cases adding risk of predation during prey relocation may result in more relocation. We also model the process by which the players can learn about the properties of the different hiding locations and find that having to learn the capture probabilities is favourable to the prey.

1.2 A Normal Form Game Model of Search and Pursuit (Chapter 3)

There is an extensive literature on search games where the prey chooses to hide at one of a finite number of locations (for example cells, boxes, etc.) and then the predator looks sequentially into these potential hiding locations to try to find the prey. These locations might be heterogeneous in the cost of searching and in the probability that a hider may escape after his location is searched. This is a simple but useful model that encompasses both the search and the pursuit portions of the predator-prey interaction. The literature on this model has been discussed in [36] with a strict assumption that all locations have the same cost to search. We relax this assumption so that each location takes a certain time to inspect and the predator has total inspection time k. We also consider a repeated game model where the capture probabilities only become known to the players over time, as each successful escape from a location lowers its perceived value capture probability.

1.3 Optimizing Voting Order on Sequential Juries: A Sealed Card Model (Chapter 4)

The celebrated Jury Theorem of [29] considered an odd size jury in which jurors vote simultaneously after receiving independent signals, correct with the same probability $p > 1/2$, concerning which of two binary possibilities (maybe innocent or guilty) is true. The reliability of the majority verdict, the probability it is correct, goes to 1 as the jury size tends to infinity. Recently, [2] considered a variation of the Condorcet model in which jurors vote sequentially (rather than simultaneously), with knowledge of prior voting. In addition, jurors are heterogeneous in a quality called ability, which affects the usefulness of their signals for predicting the true possibility. Their notion of ability was abstract, modelling for example eyesight quality for jurors making line calls in tennis. For a jury of size three of given distinct abilities, they found that reliability was order dependent and maximized when jurors vote as follows: first the juror of median ability, then of high ability, then of low ability.

This paper makes the notion of ability concrete, with a different signal distribution than Alpern-Chen. A (sealed) card is removed from a deck of m red and m black cards. Each juror draws a number of cards equal to his integer-valued ability and votes for the color of the sealed card, based on his draw and prior voting. We find that the Alpern-Chen (median-high-low) ordering has the highest reliability in most cases, and it generally is also better than simultaneous voting. In the cases where the latter is better, the abilities of the jurors tend to be similar. An analogue of this ordering has high reliability for larger juries.

1.4 The Role of Operations Research in Search - Pursuit Games and Jury Voting Problems

The subject of my thesis falls within the realm of *Operations Research.* In Search and Pursuit games, the objective of our studies is to compute the optimal search and hiding policy of a searcher and a hider. In Jury voting problem, our study focuses on using integer programming to calculate the probability of three or five jurors obtaining the final verdict which is equal to the true state of Nature. Given the heterogeneity of the jurors (each juror has different ability), our analysis aims to identify the optimal voting order: the one which gives the highest probability of reaching the final verdict equal to the true state of Nature.

Almost all of the literature published to date on search and pursuit games are known to have originated from the field of *Search Theory* [41]. The area of search theory itself is a research field within operations research which arose from research studies into air defense operations in England during World War II. The earliest conception of search theory traces back to the many problems faced by the U.S Navy in an Anti-Submarine Warfare (ASW) against the German's U-boats during the same war. In 1944, John Von Neumann developed the *Search Games* to put the findings in search theory into practices. Here it should be noted that while Operations Research and Search Games became popular around the world after WWII, the merging of the two areas has been recorded since 1951 in a well-known book by Morse and Kimball called *Methods of Operations Research* [56], which is one of the most famous Operations Research textbooks.

From the history mentioned above, it can be deduced that search game focuses on realistic problems faced by a searcher attempting to locate hider(s). The game can be defined as the study of finding an optimal ways to search for hiders in the search activity that involves searcher(s) and hider(s). Thus, one can think of search problems faced by a searcher (or symmetrically, a hider) that looks for optimal solutions in optimization problems as examples of search games.

In Chapter 2 and Chapter 3, we consider the application of search and pursuit games, using methods from stochastic games, deterministic games, as well as Markov Decision Process (MDP) to understand the behaviour of a searching predator and a hiding prey in ecological search activities. In Chapter 3 Section 3.3, we also provide a simple analysis of how the best-response problem encountered by the searcher could be viewed as a classic operations research problem, called the *Knapsack* problem.

It is not difficult to imagine our findings to be applied to many scenarios. For instance, a search and pursuit activity between a law enforcement officer and a criminal in a specific region or district. a criminal may choose to hide in a small neighborhood where the time to search is small but the probability of running away is larger due to many shortcuts and elusive nooks. Larger neighborhood on the other hand may have larger search time but higher capture probability due to the many open spaces. The problem for the searcher is to optimally distribute his search allocation to maximize the probability of finding and capturing the criminal.

In Chapter 4, we study the optimal ordering of an odd-sized jury in an open sequential vote using applications from probability techniques (similar to that of the urn sampling model), information aggregation between voters through Bayesian analysis, and optimization into group decision-making process. To illustrate how our study falls within the remit of business school particularly area of operations research, consider a scenario as follows: A firm or an organization has to make an important decision. The best decision depends on whether Nature is in one of two possible states: A or B. To determine which is the actual state, as well as possible, the organization assemble a jury of experts. For instance, if the decision is whether to put someone into jail, it convenes a legal jury to decide whether the defendant is guilty (A) or innocent (B). More often within the nature of its application, the jury may consist of economists who aim to determine the future state of the economy in order to decide whether to make an investment. In all these cases, the experts (jurors) will normally have different abilities (or expertise, judgement, economic knowledge) to determine the actual state of Nature. It is often assumed from previous works ([29], [23]) that the order in which the juror votes does not matter, given each voter can assume he is making the pivotal vote. However, when we introduce a parameter such as ability for each voter and they vote sequentially, the voting order assuredly does matter. In a three-member jury for instance, if all the three jurors have different ability, where do we put the higher–ability juror(s) in terms of voting ordering? Surely we do not want them to vote last because if the first two jurors agreed, the final verdict had been reached. If we put the highest-ability to vote first, the lower ability might disregard their own judgement and simply follow the vote of the higher–ability juror. This is called *herding.* This observation is the starting point of our analysis of how there exist a unique optimal voting ordering for heterogeneous jury who votes sequentially for binary alternatives.

Chapter 2

A Stochastic Game Model of Searching Predator and Hiding Prey

2.1 Introduction

Foraging theory is generally concerned with groups of predators and prey and considerations of spatial densities are important. However when both predator and prey density is very small, it may be a good approximation to assume that the local environment contains one predator and one prey (or none, in which case the predator is doomed anyway). In this case a two person zero sum win-lose game model may be useful, where the predator wins the local game if it finds and successfully pursues the prey and otherwise the prey wins. Such a model, with both search and pursuit considered, was introduced in [36]. In this and the following models, the prey could hide among a fixed number n of 'locations' (hiding places), and the predator had enough time or energy to look into only k of them in any period. The locations i are heterogeneous in the probability p_i that the predator successfully pursues a prey found at location i. This model was extended to multiple periods in [11] in the case that the prey is found but not caught, in which case it can relocate at any hiding place in the next period. The relocation process was assumed to be riskless for the prey. In this paper that unrealistic assumption has been relaxed in that the prey is assumed to be captured by the predator when relocating from location i to location j with known probabilities $\alpha_{i,j}$, representing the danger of such a relocation in terms of the terrain that needs to be crossed. This paper also introduces a model where the capture probabilities p_i are not know initially by the players, but are learned over time. Thus while, we model precisely the hide-seek part of the game, the pursuit part is simplified by the adoption of known values of the p_i. That part of the game might also be modeled, as in [34]. To make the relationship of the published and new models clearer, Table 2.1 compares their various properties.

	One Period	Repeated	Learning	Stochastic
symbol for game		G_k	L	Δ_k
publication reference, section	[36], 2.4.1	[11], 2.4.2	2.8	2.5, 2.6, 2.7
number of periods	1	∞	2,3	∞
searches within a period	k	k	1	k
capture between periods?	-	no	no	yes
state dependent transitions?	-	no	no	yes
discounting?	-	yes,no	no	yes,no
probabilities p_i known?	yes	yes	no	yes

Table 2.1: Comparison and Progression of the Hide-Seek-Pursuit Models

A final caution is that our notion of prey animals 'hiding' should not be taken too literally or restrictively. In fact the prey are usually carrying out some other activity, like seals choosing an ice hole for breathing [70], which they wish to do repeatedly in an unpredictable manner to avoid its predator, the polar bear. It could be choosing a water hole, as in [73]. We use the metaphor of hiding to put this problem into the hide-seek literature, which we already extended to hide-seek-pursuit. From the point of view of the predator, the location of the prey is 'hidden', not known in advance of the search procedure.

2.1.1 Qualitative summary of main results

The main results of this paper are mathematical theorems distributed throughout Sections 2.5 to 2.8. Most of these are quantitative in nature, for example we give precise optimal probabilities for the prey to hide at each location, possibly based on its prior location. However we believe it is useful to give rough qualitative versions of some of these results here. For the precise results on which these summaries are based, refer to the specific results quoted.

1. In a two location model analyzed in Section 2.7, there is only a risk of inter-period capture if the prey moves to the other location (relocates) rather than remaining at its original location. We find the counter-intuitive result that increasing the risk of relocation (higher capture probability during the move) may also increase the frequency of relocation by the prey under its optimal hiding strategy.
2. For certain data on the p_i, if the prey hides optimally in terms of the predator strategy, rather than simply hiding randomly, it can reduce the probability of eventually capture from about 0.46 to about 0.29. This is a reduction of about 37%. See equation (2.31)
3. When there is learning, higher variability of locations with respect to their capture probabilities favors the predator if these probabilities are high; but favors the prey if these probabilities are low. (Proposition 2)
4. When the patch is disrupted by some event (hurricane, drought) which may change the pursuit characteristics of the different locations, the fact that their capture probabilities must be learned again is favorable to the prey. (Proposition 1)

2.2 The Search Game Literature

The field of search games is an area of two person zero-sum games where the Hider and Searcher are in a known search region and choose their motions: the Hider (mobile or immobile) wishes to avoid or delay capture. In the search games most relevant to our model, the Hider chooses to locate at one of a finite number of locations (called cells, boxes, etc.) and then the Searcher looks sequentially into these boxes to try to find the Hider. These boxes may be heterogeneous in the overlook probability (that the Searcher looks into the correct location but does not see the Hider) and the cost of searching. The literature on this aspect of our model when the Searcher has a limited amount of time to find the Hider has been discussed in [36]. A related paper is the study of [49], who find a search strategy independent of the limited time horizon. The repetition of search in repeated periods is modeled in [7] and [8], where during the search the prey (Hider) may attempt to flee the search region. The prey will succeed in this attempt if the predator is in a cruise search mode, but not if he is in an ambush mode. In those models, a successful flight by the prey is *definitely* followed by a renewed attempt by the predator to find it. Search games with a network structure (related to transitions in our model) are studied in [18] and [15]. To extend our work to multiple hidden prey, the abstract model of [47] would be useful.

The problem of where to hide food (in discrete packages such as nuts) rather than where to hide oneself, has been analyzed in a search game played between a scatter hoarder such as a squirrel and a pilferer in [6]. The squirrel has limited digging energy and has to decide between placing nuts deeply hidden in one place or alternatively widely scattered at shallower depths. This problem is somewhat analogous to the problem of a prey hiding in a good location or randomly choosing among less good locations. Of course the payoffs are of a different kind as the prey either gets caught or not; while the squirrel either has enough nuts left to survive the winter, or not. Also, there is no pursuit phase in the squirrel's problem.

The work of [73] and [74] included ambush modes for the searching predator. A 'silent predator' (whose approach is not observable by the prey) was considered in [16]. More biologically realistic models were considered by [25] and [60]. The wider subject of search games is the subject of the monograph [9].

2.3 Behavioral Ecology Literature

The study of predator optimal foraging for stationary prey has a long history since the 60s ([69], [68]). Simple situations can be formalized using graphic methods, as for the patch leaving rule, while complex situations, as foraging in a stochastic environment, require elaborate formalism such as stochastic dynamic programming [50]. These studies show that predators follow optimal solutions, but also use simpler rules of thumb. The study of optimal escape of prey is more recent [30]. Indeed, the advent of new tracking devices, from accelerometers to UAVs, enabled the collection of massive precision data about predator and prey movements only recently (see for example [72], [24]) and the recording of the paths of both antagonists even more recently, see [27]. The field is thus currently experiencing an explosion in terms of observation and experiments, while the modeling formalism is lagging behind. Here again, several key aspects have been formalized using graphical arguments. The number of models addressing more realistic situations is however much lower. In all these approaches, the consideration is focusing on a single agent, the predator or the prey, acting in a possibly changing environment. This is the heart of optimal foraging theory.

Cases in which antagonists have no or incomplete information during the interaction have been rarely studied, both phenomenological and in terms of optimal behavior. This is surprising given their frequent occurrence in Nature. Biological examples fitting such description include polar bears hunting for seals at breathing holes, parasitic wasps hunting host larvae hidden inside leaves or wolves hunting elk in deep forests. While the first two examples have been described in detail earlier in this context ([36],[11]), the interaction between wolves and their prey was not, and is thus summarized here. The authors of [53] have patiently collated numerous observations of wolf packs pursuing many species of prey. Elk in particular (p. 68) seem to use features of the landscape to escape. They prefer areas where dead trees have toppled, creating an entanglement of logs difficult to travel through. Mountain sheep, another prey, are also unique in their agility, sure-footedness and maneuverability over rugged terrain. The hide and seek games reported in that book are great examples of search games, including the added complexity displayed by wolves, sometimes able to predict the escape route of their prey and to position themselves accordingly. As exemplified by these case studies, Nature seems replete with predator-prey interactions which are best viewed as search games. While repeated search games [11] represent the most realistic types of interactions modeled so far, they still lack essential ingredients of interactions between foraging predators and escaping prey. We focus here on two such wanting elements, the spatial distribution of risk between and among hiding sites and the change of the environment during the interaction.

The spatial distribution of the risk of predation among and between discrete hiding locations can be categorized into two extreme cases. In the first case, the locations are relatively safe places. Examples include retreat holes for mammals, hiding crevices for lizards, bushes for small passerine birds or feeding tubes for worms in the sea [30] Here, the most dangerous moments are when animals are away or out of these positions, or when they move to them. By contrast, once the retreat is reached, the probability of being caught is decreased to a large degree, at times null. In the second case, the locations represent zones of high attack probabilities, while moving between them is riskless. Breathing holes of seals attacked by polar bears or feeding windows of caterpillars attached by wasps are of this type ([36],[11]). Indeed, polar bears cannot attack seals while their travel below the ice sheet and wasps cannot attack leafminer larvae if they rest under the intact thick cuticula of a leaf. There is of course a continuum of cases spanning the two extremes. Thus, the most general model should allow capture both on site and while moving from site to site. The capture probability might be furthermore depending on the predator strategy, for example when predators choose which of the sites to visit, but might also be independent of the predator. The amount of vegetation cover, or the difficulty of progressing on the terrain between two sites are two such possible influencing factors. In these cases, there still exist site-to-site path dependent capture probabilities. We thus conclude that a realistic model should make the distinction between these risks among sites and between sites. In previous work, we dealt so far only with among site variability in predation risk. The present work is addressing both types of risks.

The search games played by foraging predators and escaping prey often unfold in conditions which usually change, possibly under the action of the players. These conditions, called environment, are here understood in a liberal fashion, being either external (time of day, for example) or internal (hunger level, for example). The proper formalism for such situations is the realm of stochastic games [67]. Furthermore, a classical optimal foraging model would not make the movement of the prey (if any) a function of the behavior of the predator. Would such two-ways interactions modify the outcome of the game? If so, in which way? These are the kind of questions we are interested in. Our aim is to develop a stochastic game framework including simultaneous decisions of two antagonists during a hide and seek game with multiple bouts in which the motivation of the predator fluctuates. This works represents therefore the natural bridge between the commonly used single predator, multiple stationary prey, optimal foraging theory described earlier and search games.

2.4 Overview of Previous Results

The current paper can be seen as an extension of [36] and [11]; we summarize those models, calling them respectively the *one stage game* and *the repeated game*. Table 2.2 explains most of the important notations used in the different models.

As related notational convention in game theory is to use "he and she" to distinguish between the two players: here we will use "he" for the Searcher and "she" for the Hider, reverting to "it" when we refer to predator and prey animals.

n	total number of locations $i, j = 1, \ldots, n$
k	number of locations searched in each period
p_i	probability of capture if prey found at location i, $p = (p_i)$
h_i	probability of prey hiding at location i (strategy), $h = (h_i)$
r_i	probability that location i is searched (strategy), $r = (r_i)$
λ	a constant equal to $\left(\sum_{i=1,\ldots,n} p_i^{-1}\right)^{-1}$
v	the game value, probability of eventual capture
β	discount factor
a, b	low and high capture probabilities in learning case
$()^*$	superscript * indicates optimal strategy

Table 2.2: List of common notations

2.4.1 The One Stage game

We now review in more detail the one stage (period) game of [36]. A (stationary) Hider locates in one of n locations $i \in \mathcal{N} = \{1, 2, ..., n\}$ while the Searcher inspects k of these, where n and k are parameters of the game. The order of inspection is not important. If the Searcher inspects the location i chosen by the Hider, the Hider is captured with a probability p_i that depends on the location i. For convenience we assume that $p_1 \leq p_2 \leq ... \leq p_n$, that is, the locations are numbered in decreasing order of attractiveness to the Hider. The Searcher wins the game if he finds and then captures the Hider. The Hider wins if she is not found or if she is found but not captured. So if the Hider hides at location i and the Searcher inspects a k-subset S (subset of cardinality k) of $\mathcal{N}$ then the payoff P to the maximizing Searcher, the probability that the Searcher wins, is given by

$$P(S, i) = \begin{cases} p_i & \text{if } i \in S, \\ 0 & \text{if } i \notin S. \end{cases} \tag{2.1}$$

If we say that the payoff to the Hider is the probability she is not found and captured, then the game has constant sum 1 (the Hider's payoff is $1 - P$).

A mixed Hiding strategy is a probability vector of hiding probabilities $h = (h_1, h_2, ..., h_n)$ where h_i is the probability that the Hider hides at location i. A mixed strategy for the Searcher is a probability distribution over k-subsets of $\mathcal{N}$. Clearly to every such mixed search strategy there is a probability r_i that location i is inspected. Conversely, if we know all the probabilities r_i, we can determine the mixed search strategy. This leads us to the following equivalent, and more useful, definition of the mixed Searcher strategy.

Definition 1. A mixed search strategy is a vector of probabilities $r = (r_1, r_2, ..., r_n)$ where r_i is the probability that the Searcher visits location i, so that

$$\sum_{i=1}^{n} r_i = k. \tag{2.2}$$

In this constant sum game, the value v is the probability of capture P, with best play on both sides. Note that if the Searcher inspects location i when the Hider is adopting the mixed strategy h, the Searcher wins with probability $h_i\, p_i$, the probability that the Hider is found multiplied by the probability she is then captured. We will often consider the mixed hiding strategy called h^* which makes all these probabilities the same, that is,

$$h_i p_i = \lambda, \text{for all } i \in \mathcal{N} \tag{2.3}$$

and for some constant λ. We say that h^* is the Hider strategy which makes all locations equally attractive for the Searcher. These equations have a unique solution given by

$$\lambda = \frac{1}{\sum_{1}^{n} \frac{1}{p_i}}, \text{ and} \tag{2.4}$$

$$h_i^* = \lambda / p_i, \ i \in \mathcal{N}. \tag{2.5}$$

It follows from the formula for λ and the assumption that the p_i are increasing in i that $1 \le p_1/\lambda \le n$.

The solution of the game is easy to see in the two extreme cases where $k = 1$ and where $k = n$. When k is 1 this is a standard hide-seek game, sometimes called a diagonal game. The value of this game is λ, the Hider should adopt h^* to make all locations equally attractive, and the Searcher should inspect locations with probabilities proportional to their capture probabilities p_i. On the other hand, when $k = n$ and all locations are inspected, only the Hider has a strategic choice and she is captured with probability p_i if she chooses location i, so clearly location $i = 1$ is best for her, with a value of p_1. The surprising finding of [36] is that for small k the solution is like that for $k = 1$ and for large k the solution is like that of $k = n$. The dividing value of k is given by p_1/λ. This result is stated below.

Theorem 1. *The solution of the one-stage game described above depends on the value of k relative to p_1/λ.*

1. *If $k < p_1/\lambda$ then the optimal hiding strategy is h^*, the optimal search strategy visits each location i with probability $r_i = k\lambda/p_i$ and the value is $k\lambda$.*
2. *If $k \geq p_1/\lambda$ then the value is p_1. The Hider can guarantee paying at most p_1 by always hiding at location 1 and the Searcher can guarantee at least p_1 by choosing $r_1 = 1 < k\lambda/p_1$ and $r_i \geq \min(k\lambda/p_i, 1)$ for all $2 \leq i \leq k$.*

This presentation of the one stage game of [36] is a good place to mention the distinction of our approach with evolutionary game theory. We note that our game is a big generalization of the so called matching pennies game, where each player chooses H or T and the maximizer wins if they choose the same and the minimizer wins if they are different. This is our game with $n = 2$ locations called H and T, with both capture probabilities equal to 1 and $k = 1$ searches. This game is mentioned in Section 4.2 of [71]. After observing that the game is not symmetric it is further observed, "Thus, matching pennies games fall outside the domain of evolutionary stability analysis." This applies equally well to our more general hide-seek-pursuit games, as well as any asymmetric game (see [64]). Obviously we cannot expect pure strategy solutions (saddle points) in hide-seek games, as certain knowledge of the hiding place ensures the prey will be found. However in matrix games such as the one presented here, iterative methods of solution are known. An evolutionary approach to search games would indeed be an interesting and useful contribution, but to our knowledge no attempts in this direction have been made, and we do not make such an attempt here. In our later stochastic game, the optimal strategies are indeed obtained by an iterative process (Corollary 5), though not exactly an evolutionary one.

2.4.2 Repeated Games

In [11], the one stage game was extended to a repeated game. We briefly review the model and results for the undiscounted and discounted versions of that game here.

The repeated game G_k

During the k looks among the **different** locations within a single patch, there can be any of the following three events:

1. If the Searcher does not find the Hider, then the game ends with zero payoff for the Searcher and a payoff of one to the Hider. (Hider wins.)
2. If the Searcher finds the Hider and captures it, then the game ends with a payoff of one to the Searcher and a payoff of zero to the Hider. (Searcher wins.)
3. If the Searcher finds the Hider but does not catch it, then the Hider escapes to another patch and the process restarts. (Game continues.)

Here the payoff P_S to the Searcher is the probability that the Hider is eventually captured (at some stage of the game). The value v of the game is obtained by solving the equation

$$\sum_1^n \frac{v}{p_i + (1 - p_i)\,v} = k. \tag{2.6}$$

The *equally attractive* hiding strategy is given by

$$h_i^* = \frac{v/k}{p_i + (1 - p_i)\,v} \tag{2.7}$$

Note that the "attractiveness" of location i in a repeated game is given by $h_i\,[p_i + (1-p_i)\,v]$. The hiding strategy h^* is optimal for all k, whereas in the one stage game it was optimal only if k was below a threshold!

The Discounted Repeated Game

The repeated game can also be studied under the assumption that the payoff is discounted by a *discount factor* β, $0 \le \beta \le 1$, in each stage. If $\beta = 0$ we have the one stage game and if $\beta = 1$ we have the undiscounted repeated game. In [11] we have shown that the value v of the discounted game is given as the unique solution of the equation

$$\sum_1^n \frac{v}{p_i + (1-p_i)\ \beta\ v} = k, \text{ when} \tag{2.8}$$

$$k \le \sum_1^n \frac{p_1}{p_i(1-\beta) + p_1\beta}, \tag{2.9}$$

and that in this case the strategy h^* is optimal for the Hider.

Otherwise, the *'stay at 1'* solution $h_1 = 1$ is optimal for the Hider, with the value

$$v = \frac{p_1}{1-\beta+p_1\beta}. \tag{2.10}$$

We have also proved the following theorem:

Theorem 2. *Consider the repeated discounted game with k looks and a discount factor β. Consider the equation (2.11).*

$$k = \sum_1^n \frac{p_1}{p_i(1-\beta_k) + p_1\beta_k}. \tag{2.11}$$

If there is a solution β_k to equation (2.11), then it is unique and

- *If $\beta < \beta_k$, then the 'stay at 1' strategy $h_1 = 1$ is the optimal strategy for the Hider.*
- *If $\beta > \beta_k$, then the **'equally attractive'** is the only solution to the game.*

If equation (2.11) has no solution in $[0, 1]$, *then the 'equally attractive' strategy* h^* *is the only optimal Hider strategy.*

2.5 The Stochastic Game Δ_k

We now present our new model. In this section we modify our repeated game model so that after a prey escapes capture at location i, she may still be captured in the course of moving to her chosen next location j (possibly the same as i if she chooses not to move between periods). We assign a fixed *probability* $\alpha_{i,j}$, which depends on the two locations, to this capture probability. The probability $\alpha_{i,j}$ is a reflection of the properties of the terrain between locations i and j. For example $\alpha_{i,j}$ might be high if the terrain in between is very open and has high visibility to the predator. In practice, this probability might depend on choices (such as where to position between periods) of the predator, but for simplicity we assume here that it is independent of any such choices. Note that if all the transition capture probabilities $\alpha_{i,j}$ are taken to be 0, then the new *stochastic game model,* which we will denote by Δ_k, reduces to the previous repeated game model G_k of ([11]).

To formally define the stochastic game $\Delta_k = \Delta_k (n, p, \alpha_{i,j})$, we must make two changes to the notation of the repeated game model. First, we need to add two additional *artificial states*, in addition to our n original location states, to indicate ending situations for the game. If the Hider has not been found at the end of the k searches allowed in a period, then the Hider wins and we say that the game moves to the artificial state $i = -1$. Alternatively, if the Searcher wins because he has found and captured the Hider, we say that the game moves to the artificial state $i = 0$. Clearly the n location states $i \in \{1, 2, \dots, n\}$ are non-absorbing (the game continues from such a state) while the two artificial states $i = -1, 0$ are absorbing states, where one of the players has won. The location state i denotes the state of the game when the Hider has been found at location i but has escaped the pursuing Searcher.

Our previous models were *constant-sum,* rather than *zero-sum* because the payoffs to the players were the probabilities that they would win the game. These probabilities sum to 1 rather than to 0. The theory of stochastic games we use here applies to zero-sum games, so we need to make a simple affine transformation of the payoffs that takes 1 to 1 and 0 to -1. (This transformation is $x \to 2x - 1$.) In the new notation the winner's payoff is $+1$ and the loser's payoff is -1, so the game is zero-sum. To transform the probability P_{S} that the Searcher wins (payoff in the repeated game) into a constant sum payoff C, we adopt the monotone increasing affine transformation given by

$$C = 2P_{\mathrm{S}} - 1. \tag{2.12}$$

Thus when the Searcher wins we have $P_{\mathrm{S}} = 1$ and $C = 1$; but when the Hider wins we have $P_{\mathrm{S}} = 0$ and hence $C = -1$. The same transformations applies as well to the values v of the repeated and stochastic games. For example a value of $v = 0$ now means that with best play either player is equally likely to win the game (the same as the value 1/2 in our previous models). Note that the probability of capture P_S satisfies

$$P_{\mathrm{S}} = (1 + C)/2. \tag{2.13}$$

The dynamics of the stochastic game Δ_k (in both the undiscounted and discounted versions) are as follows. The location state i corresponds to the situation where the Hider has been found at location i and has successfully escaped capture. Her pure choice is her next location and so her mixed choice variable at i is her distribution $h = h^i = h_1^i, \ldots, h_n^i$ over where to locate in the next period. The choice variable for the Searcher at state i consists of the k locations to search in the next period, given that the Hider has just left location i. The Searcher's mixed strategy from state i can be represented by the variable $r = r^i$, where r_j^i denotes the probability that location j is among the locations he will search. Suppose the Hider chooses location j. Then

1. She is captured before reaching the new location with probability $\alpha_{i,j}$. Otherwise,
2. With probability $1-r^i_j$ she will not be found at j, and the next state is the absorbing state -1 (Hider wins, payoff is -1).
3. With probability r^i_j she will be found at j. In this case
   (a) With probability p_j she will be captured at j and the next state is 0 (Searcher wins)
   (b) With probability $1 - p_j$ she will not be captured and will have to choose a new location for hiding. The new state is j.

2.5.1 The undiscounted and discounted stochastic games

Suppose that there exist values v_i, $i = 1, \ldots, n$, for the stochastic game when the state is a location state i, $i = 1, \ldots, n,$. It is a standard matter to find an equation which relates all the v_i. Suppose that at state i the Hider chooses to go to location j and the Searcher chooses strategy $r = r^i$. Then it is easy to see that the next state is either a location state j or one of the artificial states $-1, 0$ with the following probabilities and payoffs.

next state	probability	payoff
0 (captured in transit)	$\alpha_{i,j}$	1 (Searcher wins)
0 (found and captured at j)	$(1 - \alpha_{i,j})\ r_j\ p_j$	1 (Searcher wins)
-1 (not found at j)	$(1 - \alpha_{i,j})\ (1 - r_j)$	-1 (Hider wins)
j (found but not captured at j)	$(1 - \alpha_{i,j})\ r_j\ (1 - p_j)$	v_j (game continues)

It follows that the expected payoff if the Hider goes to location j and the Searcher uses the search strategy $r = r^i = (r_1, \ldots, r_n)$ is given by

$$\alpha_{i,j}\ (1) + (1 - \alpha_{i,j})\ \ (r_j p_j\ (1) + (1 - r_j)\ \ (-1) + r_j\ (1 - p_j)\ \ (v_j))$$

Consequently the expected payoff if the Hider adopts the mixed strategy h is given by

$$C(i, r, h, v) = \sum_{j=1}^{n} h_j \left[\alpha_{i,j} + (1 - \alpha_{i,j}) (r_j p_j - (1 - r_j) + r_j (1 - p_j) v_j) \right].$$

Theorem 3. *If the values v_i exist for all $i = 1, \ldots, n$, then they must satisfy the equations for all $i = 1, 2, ..., n$.*

$$v_i = \min_h \max_r C(i, r, h, v).$$

Existence of a value for Δ_k

The theory of stochastic games shows that the game Δ_k has a value vector $v = (v_1, \ldots, v_n)$, where v_i is the value of the game starting at location state i. A *stationary strategy* is a strategy which chooses actions depending on the current hiding place only.

The game Δ_k is two-person zero-sum with finite state and action spaces with a *positive probability to stop* for any state and any actions by the players: If the Searcher does not visit the hiding location the game stops (escape), and if the Searcher visits the hiding location the game stops (with capture) with probability at least $p_1 > 0$. Thus, we can use the fundamental result of [65] on stochastic games, that in the above mentioned conditions equilibrium exists in stationary strategies. This result is valid both for the undiscounted and the discounted stochastic games so we have the following theorem.

Theorem 4. *There exist unique values v_i , $i = 1, \ldots, n$, for the stochastic game Δ_k. This result holds both for the undiscounted and the discounted version. There exist optimal stationary strategies for both players.*

Value iteration algorithm

This algorithm has been devised by Shapley in his fundamental paper [65]. We now adapt it to the game Δ_k.

Corollary 5. *Consider the following iteration scheme, where $i = 1, 2, \ldots, n$:*

$$v_i(0) \text{ is any initial guess. Then for } L = 1, 2, \ldots, \text{ we define iteratively,} \tag{2.14}$$

$$v_i(L) = \min_h \max_r \left(\sum_{j=1}^{n} h_j \left[\alpha_{i,j} + (1-\alpha_{i,j})\left(r_j p_j - (1-r_j) + r_j(1-p_j)(v_j(L-1))\right)\right]\right). \tag{2.15}$$

Then $\lim_{L\to\infty} v_i(L) = v_i$.*This value iteration scheme converges with a geometric rate* $(1-p_1)^L$.

This algorithm works for the undiscounted and even faster for the discounted stochastic game Δ_k.

The value at the beginning

At the beginning of the game no location has been chosen yet. The Hider chooses a location i, $i = 1, ..., n$ and the Searcher chooses a set of k locations. What is the probability q_i of eventual capture in the game under the condition that the prey was discovered at location i at the first stage? With probability p_i (by definition) there is a successful pursuit and the Hider is captured. With complementary probability $1 - p_i$ the pursuit is not successful, and since the game is in state i (the Hider has escaped from location i) the definition of v_i says that the expected payoff $C = v_i$. By our affine transformation relating payoff and capture probability, equation (2.13), we have in this case that $P_{\mathrm{S}} = (1 + C)/2 = (1 + v_i)/2$. Thus overall we have

$$q_i = p_i + (1 - p_i) \times (1 + v_i)/2, \tag{2.16}$$

Thus, the game at the beginning is equivalent to the one stage game with probability of capture q_i for location i, that is, q_i plays the role of what we called p_i in the one stage game. The solution of this game is thus given by

Theorem 6. *The optimal solution of the game Δ_k can be obtained from Theorem 1 as follows:*
The capture probabilities q_i, $i = 1, ..., n$, are given by (2.16)

$$p_1 = \min_{i=1,\ldots,n} q_i. \tag{2.17}$$

and

$$\lambda = \frac{1}{\sum_1^n \frac{1}{q_i}}. \tag{2.18}$$

Then we transform the optimal probability of capture P_S into the zero-sum payoff $C = 2P_S - 1$ (see (2.12)).

2.6 A Comparative Example

We now look at the effect of both optimizing (rather than simply random) prey movement and of adding risk to inter-period prey movement (allowing $\alpha_{ij} > 0$ rather than riskless $\alpha_{ij} = 0$). We do this we by comparing our models of Section 2.4.2 (repeated games) and Section 2.5 (stochastic games) with the Markov Decision Process solution to the one sided optimization of a Searcher against a random Hider, in a simple example with just two locations.

Consider a patch with two locations ($n = 2$), $p_1 = 0.1$, $p_2 = 0.8$, and $k = 1$. In case of capture the payoff is 1 for the Searcher and 0 for the Hider and in case of ultimate escape the payoff for the Searcher (Hider) is 0 (1) so the ultimate payoff to the Searcher is the overall probability of capture. Assume that if the Hider was discovered but not captured she succeeds to reach another patch and the process continues until capture or ultimate escape. We use the undiscounted case. We denote v as the **overall probability of capture** in all the models of our toy example.

A Markov Decision Process (MDP) model for the Searcher is a framework in which his actions are optimal based on his knowledge about the current state and the strategy of the Hider. This state is fixed at the beginning but at any further stage it is the location at which the Hider was discovered (but not captured) at the previous stage. We now compare the MDP to the stochastic game version of this model. At first we neglect the capture risk of the Hider during changing locations, and then we take this risk into consideration. Then we consider the possibility of inter period capture.

2.6.1 Model without risk when changing location

We now consider the earlier model where between periods the Hider prey can move between locations without risk of capture, so that all the transition capture probabilities α_{ij} are 0. We first consider that the prey acts randomly and then considers that the prey acts so as to minimize capture probability. In both cases, we assume that the searching predator acts to maximize capture probability.

Random prey, optimizing predator (MDP model)

Assume that the Hider always hides randomly and uniformly, i.e., $h = (0.5, 0.5)$, and that there is no risk for the Hider to change locations. That is, the Hider equiprobably stays or changes location between periods. In this case the optimal strategy of the Searcher is to always look at location 2 (or adopt strategy $R = (0, 1)$ in our notation) at each stage.

The ultimate probability of capture v satisfies

$$v = .5 \times (.8 + .2 \times v) + 0, \text{ giving } v = \frac{.4}{.9} \simeq\ 0.44 \tag{2.19}$$

Optimal play in repeated game model

Here also we assume no risk to change locations, so this is the repeated game Δ. The optimal strategies are for both players to hide/search at location 1 with probability about 0.73 and at location 2 with probability about 0.27. The equation of the value of the game is given by equation (2.6), so we have

$$\frac{v}{p_1 + (1 - p_1)\, v} + \frac{v}{p_2 + (1 - p_2)\, v} = 1, \text{ since } n = 2,\ k = 1. \text{ So} \tag{2.20}$$

$$\frac{v}{.1 + .9v} + \frac{v}{.8 + .2v} = 1, \text{ giving } v \simeq\ 0.22 \tag{2.21}$$

Thus by hiding at the better location (location 1) with higher probability (.73 rather than .50) the Hider prey reduces the probability of eventual capture from about .44 to about .22, that is, by about 50%.

2.6.2 Model with risk when Hider changes locations

We now make the main assumption of this paper, that the Hider can be captured between periods when moving from location i to location j, with a possibly positive probability α_{ij}. For this example we make the simple assumption that the Hider cannot be captured if she stays at the same location, $\alpha_{ii} = 0$ for $i, j = 1, 2$, but that any move between distinct locations has capture probability 0.3, that is $\alpha_{ij} = .3$ for $i \neq j$. State $i = 1, 2$ corresponds to the event that the Hider has been discovered but not caught at location i and v_i is the **overall probability of capture** at that state.

Hider moves randomly, Searcher optimizes (MDP model)

We assume that from any state $i = 1, 2$, the Hider moves equiprobably to either location and that the Hider starts equiprobably at either location (not equiprobably in either state). Clearly in this case the Searcher should always look at location 2, where he has a higher chance of capturing the prey if she is there. We therefore have the following equations for v_1 and v_2,

$$v_1 = .5 \times 0 + .5 \times [.3 + .7 \times (.8 + .2v_2)] \tag{2.22}$$

$$v_2 = .5 \times (.8 + .2v_2) + .5 \times .3, \text{ giving} \tag{2.23}$$

$$v_1 \simeq 0.47, \; v_2 \simeq .61 \tag{2.24}$$

If the Hider starts in location 1, she will not be found and so the payoff is 0. If she starts at location 2, she will be found and she will be captured with probability $p_2 = .8$. She will not be captured with probability .2 in which case the eventual capture probability is v_2. So overall the capture probability in this scenario at the beginning of the game is given by

$$v = .5 \;\; (0) + .5 \;\; (.8 + .2 \; v_2) \simeq 0.46 \tag{2.25}$$

Both Hider and Searcher optimize (stochastic game model)

We now analyze the model of this paper, covering the scenario with inter-period capture risk and two optimizing players in a stochastic game. State $i = 1, 2$ corresponds to the event that the Hider has been discovered but not caught at location i and v_i is the **overall probability of capture** at that location (this is different from the notation in chapter 5). For the stochastic game we have the following equations, where the minimum is with respect to the Searcher looking at location 1 (left) or location 2 (right):

$$v_1 = \max_{0 \le x \le 1} \min [x(.1 + .9v_1) + (1 - x)\,(.3)\,; (1 - x)\,(.3 + .7(.8 + .2v_2))] \tag{2.26}$$

$$v_2 = \max_{0 \le y \le 1} \min [y(.8 + .2v_2) + (1 - y)\,(.3)\,; (1 - y)\,(.3 + .7(.1 + .9v_1))], \tag{2.27}$$

where x and y are the probabilities that the Hider will stay at the same node after escaping capture at locations 1 and 2 respectively. The solution is $v_1 \simeq .39$ and $v_2 \simeq .45$, with $x \simeq .60$, $y \simeq .27$. Note that x, the probability to stay at location 1, is smaller than the corresponding result in the model without risk in moving. This is, obviously, counter intuitive and will be later explained in the Section 2.7.

At the beginning of the game, the overall probabilities of eventual capture if the Hider is discovered at location 1 is

$$q_1 = p_1 + (1 - p_1) \times v_1 = .44 \tag{2.28}$$

and at location 2 is

$$q_2 = p_2 + (1 - p_2) \times v_2 = .89. \tag{2.29}$$

The optimal hiding policy at the beginning , as given by Theorem 1 case 1, is about $\left(\frac{2}{3}, \frac{1}{3}\right)$, the same as the optimal search strategy for the first stage. Thus the overall probability of capture, since both must go to the same location, is given by

$$v = \left(\frac{2}{3}\right)^2 q_1 + \left(\frac{1}{3}\right)^2 q_2 = .29, \text{ which can also be obtained from Theorem 6 as} \tag{2.30}$$

$$\lambda = \frac{1}{\frac{1}{.44} + \frac{1}{.89}} = .29. \tag{2.31}$$

The Hider thus reduces the probability of capture from about 0.46 for the random strategy to about 0.29 when playing optimally in the stochastic game. This is a reduction of about 37% if she uses the optimal hiding strategy, a function of the predator's actions rather than a random choice of locations. Either if we neglect the risk of moving, or if we take it into account, there is thus a marked difference in the probabilities of capture and escape between the stochastic and the single agent games, as used in most optimal foraging theory.

2.7 Relocation Probability and Relocation Risk

In Section 2.6 we noted the seemingly counter intuitive numerical result that the probability of moving increased when such a move became more risky. We now present a very simple numerical example that will enable us to understand why it happens. Assume we have two locations with probability of capture $p_1 = p_2 = 1 - \varepsilon$ and $k = 1$. Consider first the repeated game with no risk of moving. The optimal strategy for the Hider is always to hide at each location with probability 1/2. Now consider the same example with risk $\alpha = 1/2$ for changing location. If the Hider has been discovered but not captured then it is easy to see that she should make both locations equally attractive for the Searcher so she chooses the probability to stay at her present location equal to 1/3. This means she will be captured in transit with probability $(2/3)\,\alpha = 1/3$, she will be at location 1 in the next period with probability 1/3, and she will be at location 2 with probability $(2/3)\ (1 - \alpha) = 1/3$. So, conditional on her still playing the game, she is equally likely to be at either location. This choice guarantees her to lose the game with probability about 2/3 which is the minimum possible, while staying with probability 1/2 leads to losing with probability 3/4. The paradox is that we have the same (simple) model but *increasing* the risk of moving also *increases the probability of moving*.

We now give an example which makes this phenomenon simpler, without any numbers. We consider the following general case of two identical locations with a common capture probability $p = p_1 = p_2 < 1$. We suppose that staying still is safe ($\alpha_{11} = \alpha_{12} = 0$) and relocating either way has the same probability α of being captured. The symmetry of the two locations ensures that $v_1 = v_2 = v$. From state 1 (after a successful escape at location 1) the game matrix is as follows:

	stay at 1	move to 2
look in 1	$p+(1-p)\,v$	a
look in 2	0	$a+(1-a)\,(p+(1-p)\,v)$

Payoff Matrix when Hider starts at location 1

The existence of a value for this game, which we denote by v, follows from Shapley's result, our Theorem 4. First note that there is no pure strategy equilibrium. Suppose the Hider stays at location 1 with probability q, moving to location 2 with complementary probability $1-q$. The equation obtained by equating the payoffs (eventual capture probabilities) when the Searcher looks at location 1 (top row, left side of equation) and location 2 (bottom row, right side of equation) is given by

$$q\,(p+(1-p)\,v)+(1-q)\,(a)=q\,(0)+(1-q)\,(a+(1-a)\,(p+(1-p)\,v))\,,\ \text{or}$$
$$q\,(p+(1-p)\,v)=(1-q)\,(1-a)\,(p+(1-p)\,v)\,,\text{or}$$
$$q=(1-q)\,(1-a)\,,\ \text{(independent of } v\text{) with solution}$$
$$q=\frac{1-a}{2-a},\ \text{which is decreasing in } a. \tag{2.32a}$$

Note that the optimal probability q of staying at 1 (or at 2, by symmetry) does not depend on the common capture probability p or the common value v. The optimal probability is $q=1/2$ when there is no relocation risk ($a=0$). This makes sense because it makes the Hider distribution most random. The optimal probability $1-q$ of relocating is given by

$$1-q=1-\frac{1-a}{2-a}=\frac{1}{2-a},\ \text{which approaches 1 as } a\to 1.$$

We note that this symmetric model extends easily to n identical locations, where by symmetry of locations the Hider has the two choices: remain at her current location or move to a randomly selected new location. For n such locations the formula for remaining becomes $q = (1 - a)/(n - a)$.

Thus the prey may have more incentive to relocate when this move becomes riskier. We note that a somewhat similar observation was made, in a slightly different context, in [7] and [8]. There, a prey had to decide when to change locations when facing a predator who might either be in cruising search mode or in ambush mode. If the predator was in ambush mode then changing locations resulted in capture. However it was found that as the unsearched region decreased in size, the predator was more likely to be in ambush mode (so a higher "relocation cost" for the prey), but nevertheless the predator optimally increased her likelihood of relocating. The specifics of the calculations are different than those given here, as the model is only partly similar. The idea is that the relocation cost α in the current model has some similarity to the ambush frequency in the earlier papers in that both incur a risk to a prey who changes location. It would be useful to have an additional explanation for the counter intuitive result that could be put purely in words, without the necessity of a mathematical model.

2.8 Learning the Capture Probabilities

It is a natural question to ask how the predator and prey know the capture probabilities p_i; can they be learned? To answer this question we give a simple learning model. We consider the simplest case that allows for learning: two locations and two (or more) periods and only $k = 1$ location to be searched in each period. We assume that at each location the capture probabilities are known to be a or b equiprobably and independently, with $a < b$. (If $a = b$ there is nothing to be learned.) This means locations either have a low capture probability or a high capture probability, only it is not known which. At a location where the prey has escaped j times, the conditional probability that the capture probability is a (low) is denoted by $g(j)$, where $g(0) = 1/2$ and by Bayes Law,

$$g(j) = \frac{(1-a)^j}{(1-b)^j + (1-a)^j}, \text{ with } \lim_{j\to\infty} g(j) \nearrow 1. \tag{2.33}$$

In other words, each successful escape from a location makes it more likely that it has a low capture probability and hence makes it more attractive to the prey and hence also to the predator. The *effective capture probability*, denoted $\hat{p}$, is initially given simply by $\hat{p}(0) = (a+b)/2$, and more generally by

$$\hat{p}(j) = g(j)\ a + (1 - g(j))\ b = \frac{a\ (1-a)^j + b\ (1-b)^j}{(1-a)^j + (1-b)^j} \searrow a \tag{2.34}$$

For example, if $a = 1/3$ and $b = 2/3$, then the first values of $g(j)$ given by 1/2, 2/3, $4/5, \ldots, 2^j/\left(2^j+1\right)$ and the $\hat{p}(j)$ by 1/2, 4/9, .400, .370, 0.353, 0.343, 0.338.

Note that when only one location is searched in each period, the payoff matrix has 0 entries off the diagonal (when Hider is not in the searched location) so the matrix is a diagonal matrix. For two locations this is a matrix of the form

$$\begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}, \text{ with value } \lambda(d_1, d_2) = (1/d_1 + 1/d_2)^{-1} \tag{2.35}$$

and optimal probability (for both players) of strategy i is given by $\lambda(d_1, d_2)/d_i$. So in the final period of a game, if there have been i escapes from location 1 and j escapes from location 2, the payoff matrix of this one stage game, called $L_{i,j,1}$, is simply $\lambda(\hat{p}(i), \hat{p}(j))$. More generally let $L_{i,j,m}$ be the learning game where location 1 has had i escapes, 2 has had j escapes, and there are m more plays of the game. These games are recursively described by the matrix

$$L_{i,j,m} = \begin{pmatrix} \hat{p}\,(i) + (1 - \hat{p}\,(i)) \;\; L_{i+1,j,m-1} & 0 \\ 0 & \hat{p}\,(j) + (1 - \hat{p}\,(j)) \;\; L_{i,j+1,m-1} \end{pmatrix}, \text{ with value}$$

$$v\,(i,j,m) = \lambda\,(\hat{p}\,(i) + (1 - \hat{p}\,(i)) \quad v(i+1,j,m-1))\,, \hat{p}\,(j) + (1 - \hat{p}\,(j)) \;\; v(i,j+1,m-1),$$

In the game $L_{i,j,m}$ it is easy to show that the optimal probability of hiding/searching in location 1 is given by

$$x_{i,j,m} = v\,(i,j,m)\,/\,(\hat{p}\,(i)) + (1 - \hat{p}\,(i)) \;\; v\,(i+1,j,m-1)\,. \tag{2.36}$$

For our example $a = 1/3$, $b = 2/3$, the values $v\,(0,0,m)$ for games with $m = 1, \ldots, 6$ periods are given by

$$\lambda\,(1/2, 1/2) = \frac{1}{1/\,(1/2) + 1/\,(1/2)} = 0.25, \frac{21}{68} \simeq 0.309, 0.324, 0.327, 0.328, 0.32895, 0.3290 \tag{2.37}$$

For the two-stage game $m = 2$, the players randomize between the symmetric locations 1 and 2 in the first period, and assuming we name the location of escape in the first period as location 1, they go back to the same location in the second period with probability $x_{1,0,1} = 9/17 > 1/2$. Now suppose that there are three stages. Clearly in the first stage the players have no choice but to locate equiprobably to the two locations. But how do they play in the second stage (assume there was an escape at location 1) if they know it is a three stage game? In this case the probability of choosing location 1 (for both hiding and searching) is given by $x_{1,0,2} = 99/191 < 9/17 = x_{1,0,1}$. This says that the presence of an additional final (third) period decreases the probability of going back to the same location as the escape in the previous period, but this probability is still greater than one half. In fact we find this phenomenon is true in general, learning reduces the bias toward returning to locations one has escaped from. This phenomenon obviously requires three stages in our model. Using numerical methods, this can be shown to be true for all a and b.

It is useful to compare the learning game with low and high capture probabilities a and b with the similar non-learning game with a fixed and known capture probability $(a+b)/2$ which is the effective capture probability of the learning game. We consider both in the setting of a two stage game with identical locations. If the capture probability in the non learning game is c at both locations, then the value of the second stage is given by $\lambda(c,c) = 1/(1/c + 1/c) = c/2$ and hence in the first stage has value $(1/2)(c \cdot 1 + (1-c)\ (c/2)) = (3-c)\,c/4$ (half the time they go to the same location, capture (payoff 1) has probability c and escape (with payoff $c/2$ from previous calculation) has payoff $c/2$. For the example $a = 1/3$, $b = 2/3$, $c = 1/2$ the non learning game has value $(3 - 1/2)(1/2)/4 = 5/16 = 0.3125$ while the learning game has the lower value $v_{0,0,2} = 21/68 \simeq 0.3088$. This means that the capture probability is lower (better for the Hider) when there is learning. We show that this observation holds in general, at least for the two stage game.

Proposition 1. *When there are two stages and identical locations (a priori), the optimal capture probability (value) is lower in the learning game with capture probabilities $a < b < 1$ than in the game where the fixed capture probability is set equal to the effective capture probability $\hat{p}(0) = c = (a+b)/2$. That is, $v(0,0,2) < (3-c)\,c/4$.*

Proof. After some algebraic simplification, the difference in the values between the no learning and learning games can be written as

$$\frac{(3-c)\,c}{4} - v(0,0,2) = \frac{(a-b)^2\ (2-a-b)\ (a+b)}{16\ (3(a-a^2) + 3(b-b^2) + 1(a+b-2ab))} > 0, \qquad (2.38)$$

because all the factors in the numerator are positive, and for the denominator we note that $a > a^2$, $b > b^2$ and $a + b > 2a > 2ab$. $\square$

An interesting question concerns the variability of the capture probabilities. For example in a final period, is it better for the Hider to have escaped twice from one location (and face varied capture probabilities $\hat{p}\,(2)$ and $\hat{p}(0) = (a+b)/b$) or once from each location (with an effective capture probability $\hat{p}\,(1)$ at each location)? In other words, what is the sign of $v(2,0,1) - v(1,1,1)$? It turns out that the answer depends in a simple way on the size of the two probabilities a and b. If $a + b > 1$, then the Hider prefers the low variability case $L_{1,1,1}$; if $a+b < 1$ the Hider prefers the high variability case $L_{2,0,1}$; if $a + b = 1$ the players are indifferent between these cases. In particular we have the following.

Proposition 2. *Highly varying capture probabilities favor the Searcher if these probabilities are high; otherwise they favor the Hider. In particular, the sign of* $v(2,0,1) - v(1,1,1)$ *is positive if* $a + b > 1$*; negative if* $a + b < 1$ *and zero if* $a + b = 1$.

Proof. The difference $v(2,0,1) - v(1,1,1)$ is given by the fraction

$$\frac{(a-b)^4\,(a+b-1)}{2\,(2-a-b)\left(3a^3 + a^2\,(b-6) + a\,(b-2)^2 + b\,(4-6b+3b^2)\right)}. \tag{2.39}$$

The multinomial in the denominator has a minimum of 0 at $a = b = 1$, so for $0 < a < b \leq 1$ the sign of the fraction is the sign of $a+b-1$, as claimed. (Note that if we allowed $a = b$ then the escapes are irrelevant to current probabilities and the difference would also be 0.) $\square$

A normal form game (not a dynamic game) which considers this type of learning was analyzed in [14].

2.9 Discussion

Biology, economics, computer science and studies of human behavior have since long considered stochastic games ([26],[35]). A recent important perspective is presented by [67]. What is therefore new here is application to the context of search games and behavioral ecology. Hence, our work has implications beyond behavioral ecology for any situation described by hide and seek games, from ecology, immune systems to computer science [36]. We are now in the position to assess the change in success of attack and escape in stochastic search games, compared to the situation in which one player is against another one which is moving randomly. This later case is known to be equivalent of games in which only one player is behaving optimally, a Markov Decision Process (MDP), see [61] or [62]. Of course it is necessary to observe that our modelling is appropriate only in the case of low densities for both predator and prey, allowing each to assume that there is at most one of the opposite type in the search region.

While we have given a complete solution to these problems in the text, the specific example of Section 2.6 is sufficient to indicate some differences in the capture time for differing assumptions. See Table 2.3.

	capture on transition	no transition capture	% reduction
random prey, MDP	.46	.44	4%
optimizing prey, Game	.29 (stochastic game)	.22 (repeated game)	24%
% reduction	37%	50%	

Table 2.3: Expected eventual capture probabilities v for various assumptions

It is of course obvious that optimizing prey do better (lower v) than random prey and that prey would benefit from having a risk free transition between locations between periods. In general, the prey reduces the capture probability by about 37-50% if she uses the optimal hiding strategy rather than moving and/or hiding randomly. There is thus a marked difference in the probabilities of capture and escape between our (repeated or stochastic) game theoretic models and single agent predator optimization models, as used in most optimal foraging theory. This marked difference extends to the use of space by the protagonists, as in our no transition capture example of Section 2.6 (middle column) the predator should always visit location 2 in the first case, and should concentrate its visits on the first location in the second case: a complete reversal of distribution of effort as function of the tightness of the interaction!

These differences are the explanation why organisms tend to act according to the other player's actions and why the stochastic/repeated search game approach supersedes the classical optimal foraging one for modelling such interactions: the more complex modelling approach reflects the complex, multi-step trajectories of the antagonists as we observe them. The myriad of delicate and intricate biochemical, physiological or behavioral adaptations of prey for escaping predators and of predators for successfully attacking and subduing their prey [66] show that natural selection is acting on all these traits. A stochastic game formulation is thus definitely required when players do behave according to what the other is doing. The model can be developed in two promising directions. First, we did not consider prey fatigue or more complex situations in which prey balance risk of predation with risk of starvation. A refined model taking fatigue into account would have then three state variables – motivation, fatigue, and the recent location of encounter and its development would follow lines similar as the ones we have proposed.

Finally, our model is a zero-sum game. One may argue that a real game between a predator and a prey is not a zero-sum game, as the predator is running after its dinner while the prey is running for its life. This is an important if difficult aspect to deal with. Indeed, while non-zero sum stochastic games have been modeled only a few years after zero-sum games were developed, the level of complexity is strongly increased. The value of the game cannot indeed be taken as granted, in contrast to zero-sum games. For search games, implementing non-zero sum games represents a virgin and much needed field. We advocate future analysis of the following non-zero sum model. In the single period problem we could require the predator to search the k locations sequentially. If the prey is found on the j'th search and successfully pursued the payoff to the predator would be $1 - jc$ for some small fixed search cost c, modeling the effort or energy of a search. The cost could also be location dependent, c_i. The prey also might prefer later capture within a period in such a model, but this would still not make it zero-sum, as survival would be more significant.

Chapter 3

A Normal Form Game Model of Search and Pursuit

3.1 Introduction

Traditionally, search games and pursuit games have been studied by different people, using different techniques. Pursuit games are usually of perfect information and are solved in pure strategies using techniques involving differential equations. Search games, on the other hand, typically require mixed strategies. Both Pursuit and Search games were initially modelled and solved by Rufus Isaacs in his book [42]. The first attempt to combine these games was the elegant paper of Gal and Casas [36]. In their model, a hider (a prey animal in their biological setting) begins the game by choosing among a finite set of locations in which to hide. The searcher (a predator) then searches (or inspects) k of these locations, where k is a parameter representing the time or energy available to the searcher. If the hiding location is not among those inspected, the hider wins the game. If the searcher does inspect the location containing the hider, then a pursuit game ensues. Each location has its own capture probability, known to both players, which represents how difficult the pursuit game is for the searcher. If the search-predator successfully pursues and captures the hider-prey, the searcher is said to win the game. This is a simple but useful model that encompasses both the search and the pursuit portions of the predator-prey interaction.

This paper has two parts. In the first part, we relax the assumption of Gal and Casas that all locations are equally easy to search. We give each location its own search time and we give the searcher a total search time. Thus he can inspect any set of locations whose individual search times sum to less than or equal to the searcher's total search time, a measure of his resources or energy (or perhaps the length of daylight hours, if he is a day predator). We consider two scenarios. The first scenario concerns n hiding locations, in which the search time at each location is inversely proportionate with the capture probability at that location. In the second, we consider that there are many hiding locations, but they come in only two types, identifiable to the players. Locations within a type have the same search time and the same capture probability. There may be any number of locations of each type.

The second part of the paper relaxes the assumption that the players know the capture probability of every location precisely. Rather, we assume that a distribution of capture probabilities is known. The players can learn these probabilities more precisely by repeated play of the game. We analyze a simple model with only two locations and two periods, where one location may be searched in each period. While simple, this model shows how the knowledge that the capture probabilities will be updated in the second period (lowered at a location where there was a successful escape) affects the optimal play of the game.

3.2 Literature Review

An important contribution of the paper of Gal and Casas discussed in the Introduction is the analysis involves finding a threshold of locations beyond which the searcher can inspect. If this is sufficiently high, for example if he can inspect all locations, then the hider adopts the pure strategy of choosing the location for which the probability of successful pursuit is the smallest. On the other hand, if k is below this threshold (say $k = 1$), the hider mixes his location so that the probability of being at a location multiplied by its capture probability (the desirability of inspecting such a location) is constant over all locations.

The paper of Gal and Casas [36] requires that the searcher knows his resource level (total search time) k. In a related but not identical model of Lin and Singham [49] it is shown that sometimes the optimal searcher strategy does not depend on k. This paper is not directly related to our findings but the reader may find it useful to know the distinction between this paper and ours.

Alpern, Gal, and Casas [11] extended the Gal-Casas model by allowing repeated play in the case where the searcher chose the right location but the pursuit at this hiding location is not successful. They found that the hider should choose his location more randomly when the pursuing searcher is more persistent.

More recently, Hellerstein, Lidbetter, and Pirutinsky [48] introduced an algorithm similar to that of the *fictitious* play where the searcher recursively updates his optimal strategy after knowing the response of the opponent's. They apply this technique to games similar to those we consider here. Their algorithm is likely to prove a powerful technique for solving otherwise intractable search games.

More generally, search games are discussed in Alpern and Gal [10] and search and pursuit problems related to robotics are categorized and discussed in Chung, Hollinger and Isler [28].

3.3 Single Period Game with General Search Times

Consider a game where the searcher wishes to find the hider at one of n locations and then attempt to pursue and capture it, within a limited amount of resources denoted by k. Each location i has two associated parameters: a *search time* t_i required to search the location and a *capture probability* $p_i > 0$ that if found at location i the searcher's pursuit will be successful. Both t_i and p_i are known to the searcher and the hider.

The game $G(n, t, p, k)$, where $t = (t_1, \ldots, t_n)$ and $p = (p_1, \ldots, p_n)$ represent the time and capture vectors, is played as follows. The hider picks a location $i \in N \equiv \{1, 2, ..., n\}$ in which to hide. The searcher can then inspect search locations in any order, as long as their total search time does not exceed k. The searcher wins (payoff = 1) if he finds and then captures the hider; otherwise the hider wins (payoff = 0). We can say that this game is a constant sum game where the value $V = V(k)$ is the probability that the predator wins with given total search time k.

A mixed strategy for the hider is a distribution vector $h \in H$, where

$$H = \left\{ h = (h_1, h_2, \ldots, h_n) : h_i \geq 0, \sum_1^n h_i = 1 \right\}.$$

A pure strategy for the searcher is a set of locations $A \subset N$ which can be searched in total time k. His pure strategy set is denoted by $a(k)$, where

$$a(k) = \{A \subset N : T(A) \equiv \sum_{i \in A} t_i \leq k\}.$$

The statement above simply states that a searcher can inspect any set of locations for which the total search time does not exceed his maximum search time k. A mixed search strategy is a probabilistic choice of these sets.

The payoff P from the perspective of the maximizing searcher is given by

$$P(A,i)=\begin{cases} p_i & \text{if} \quad i\in A, \text{ and} \\ 0 & \text{if} \quad i\notin A. \end{cases}$$

As part of the analysis of the game, we may wish to consider the best response problem faced by a searcher who knows the distribution h of the hider. The "benefit" of searching each location i is given by $b_i = h_i\ p_i$, the probability that he finds and then captures the hider (prey). Thus when h is known, the problem for the searcher essentially is to choose the set of locations $A \in \alpha(k)$ which maximizes $b(A) = \sum_{i\in A} b_i$. This is a classic Knapsack problem from the Operations Research literature (A seminal book of the Knapsack problem is by Kellerer, Pferschy and Pisinger [45]). The *objects* to be put into the *knapsack* are the locations i. Each has a 'weight' t_i and a benefit b_i. He wants to fill the knapsack with as much total benefit subject to a total weight restriction of k.

The knapsack approach illustrates a simple domination argument: the searcher should never leave enough room (time) in his knapsack to put in another object. However to better understand this observation, we show the definition of *Weakly dominant* below

Definition 2. Strategy X weakly dominates strategy Y iff (I) X never provides a lower payoff than Y against all combinations of opposing strategies and (II) there exists at least one combination of strategies for which the payoffs for X and Y are equal.

Having stated this, we write this simple observation as follows.

Lemma 7. *Fix k. The set $A \in \alpha(k)$ is weakly dominated by the set $A' \in \alpha(k)$ if $A \subset A'$ and there is a location $j \in A' - A$.*

Proof. If i is in both A or i is not in A', then $P(A,i) = P(A',i)$. If $i \in A' - A$ then $P(A',i) = p_i > 0 = P(A,i)$. $\square$

3.3.1 An example

To illustrate the general game we consider an example with $n = 4$ locations. The search times are given by $t = (5, 3, 4, 7)$ and the respective capture probabilities are given by $p = (.1, .2, .15, .4)$. In this example it is easiest to name the locations by their search time, so for example the capture probability at location 7 is 0.4. The searcher has total search time given by $k = 7$, so he can search any single location or the pair $\{3, 4\}$. The singleton sets $\{3\}$ and $\{4\}$ are both dominated by $\{3, 4\}$. We put the associated capture time next to the name of each location. Thus the associated reduced matrix game is simply

A\location	5 (.1)	3 (.2)	4 (.15)	7 (.4)
$\{5\}$	.1	0	0	0
$\{7\}$	0	0	0	.4
$\{3, 4\}$	0	.2	.15	0

Solving the matrix game using online solver (Avis, et.al. [17]) shows that the prey hides in the four locations with probabilities $(12/23, 0, 8/23, 3/23)$ while the searcher inspects $\{5\}$ with probability 12/23, $\{7\}$ with probability 3/23, and $\{3, 4\}$ with probability 8/23. The value of the game, that is, the probability that the predator-searcher finds and captures the prey-hider, is 6/115. Our approach in this paper is not to solve games in the numerical fashion, but rather to give general solutions for certain classes of games, as Gal and Casas did for the games with $t_i = 1$.

3.3.2 The game with t_i constant

Choosing all the search times t_i the same, say 1, we may restrict k to integers. This is the original game introduced and solved by Gal and Casas [36]. Since the t_i are the same, we may order the locations by their capture probabilities, either increasing or decreasing. Here we use the increasing order of the original paper. Clearly if $k = 1$ the hider will make sure that all the locations are equally good for the searcher ($p_i h_i =$

constant) and if $k = n$ the hider knows he will be found so he will choose the location with the smallest capture probability (here location 1). The nice result says that there is a threshold value for k which divides the optimal hiding strategies into two extreme types.

Proposition 3 (Gal-Casas [36])**.** *Consider the game $G(n, t, p, k)$ where $t_i = 1$ for all i and the locations are ordered so that $p_1 \leq p_2 \leq \cdots \leq p_n$. Define $\lambda = \sum_{i=1}^{n} 1/p_i$. The value of this game is given by $\min(k\lambda, p_1)$. If $k < p_1/\lambda$ then the unique optimal hiding distribution is $h_i = \lambda/p_i$, $i = 1, \ldots, n$. If $k \geq p_1/\lambda$ then the unique optimal hiding strategy is to hide at location 1.*

3.3.3 The game with $t_i = i$, p_i decreasing, $k = n$ odd.

We now consider games with $t_i = i$ and p_i decreasing. In some sense locations with higher indices i are better for the hider in that they take up more search time and have a lower capture probability. Indeed if the searcher has enough resource k to search all the locations ($k = \sum_{i=1}^{n} t_i = n(n+1)/2$) then of course the hider should simply hide at location n and keep the value down to p_n. Note that if $k < n$, the hider can win simply by hiding at location n, which takes time $t_n = n$ to search. When $k = 0$, then the searcher cannot inspect any location and so the hider wins (Payoff = 0). We give a complete solution for the smallest nontrivial amount of resources (total search time) of $k = n$. Let us first define the following two variables which will be widely used in our main result.

$$S(p) = \textstyle\sum_{j=m+1}^{2m+1} 1/p_j; \;\; \bar{h}_j = 1/(p_j S(p)).$$

Proposition 4. *Consider the game $G(n, t, p, k)$, where $t_i = i$, p_i is decreasing in i and $k = n = 2m + 1$. Then*

1. *An optimal strategy for the searcher is to choose the set $\{j, n - j\}$ with probability $1/(p_j \; S(p))$ for $j = m + 1, \ldots, n$.*

2. *An optimal strategy for the hider is to choose location j with probability $\bar{h}_j$ for $j \geq m+1$ and not to choose locations $j \leq m$ at all.*

3. *The value of the game is $V = \frac{1}{S(p)}$.*

Proof. Suppose the searcher adopts the strategy suggested above. Any location i that the hider chooses belongs to one of the sets of the form $\{j, n-j\}$ for $j = m+1, \ldots, n$, where the set $\{n, 0\}$ denotes the set $\{n\}$. Since for $j \geq m+1$ we have $j > n-j$ and the p_i are decreasing, the hider is better off choosing location j. In this case he is found with probability $1/(p_j\ S(p))$ and hence he is captured with probability at least $p_j\ (1/(p_j\ S(p))) = 1/S(p)$.

Suppose the hider adopts the hiding distribution suggested above. Note that no pure search strategy can inspect more than one of the locations $j \geq m+1$. Suppose that location j is inspected. Then the probability that the searcher finds and captures the hider is given by $\bar{h}_j p_j = 1/(p_j S(p))\ p_j = 1/S(p)$. It follows that $S(p)$ is the value of the game. $\square$

It is natural to also analyse if Proposition 4 still holds true for $k = n =$ even number. For the simplicity of our notation and better readability of Proposition 4, we decided to write this separate section for even number. In the case where $k = n =$ even, the solution is exactly the same as their odd counterpart. More specifically $k = n = 2m$ has the same value and optimal strategies as $k = n = 2m+1$. However, it is important to note that in the even case, both the searcher's and hider's optimal strategy is unique. For instance, $k = n = 4$ has the same value and optimal strategies as $k = n = 5$. The same can be said for 6 and 7, 8 and 9, etc.

Corollary 8. *Assuming the p_i are strictly decreasing in i, the hider strategy $\bar{h}$ given above is uniquely optimal, but the searcher strategy is not.*

Proof. Let $h^* \neq \bar{h}$ be a hiding distribution. We must have $h_j^* + h_{n-j}^* > \bar{h}_j + \bar{h}_{n-j} = 1/(p_j S(p))$ for some $j \geq m+1$; otherwise the total probability given by h^* would be less than 1. Against such a distribution h^*, suppose that the searcher inspects the two locations j and $n-j$. Then the probability that the searcher wins is given by $p_j h_j^* + p_{n-j} h_{n-j}^* \geq p_j \left(h_j^* + h_{n-j}^*\right)$ because $p_j < p_{n-j}$. But by our previous estimate $h_j^* + h_{n-j}^* > 1/\left(p_j S\left(p\right)\right)$ this means the searcher wins with probability at least $p_j \left(1/\left(p_j S\left(p\right)\right)\right) = 1/S\left(p\right)$ and hence h^* is not optimal.

Next, consider the searcher strategy which gives the same probability as above for all the sets $\{j, n-j\}$ for $j \geq m+2$ but gives some of the probability assigned to $\{m+1, m\}$ to the set $\{m+1, m-1\}$. Let's say the probability of $\{m+1, m-1\}$ is a small positive number ε. The total probability of inspecting location $m+1$ (and all larger locations) has not changed. The probability of inspecting location m has gone down by ε. So the only way the new searcher strategy could fail to be optimal is potentially when the hider chooses location m. In this case the probability that the searcher wins is given by

$$\left(\left(1/\left(p_{m+1}\ S\left(p\right)\right)\right) - \varepsilon\right)\ p_m.$$

Comparing this to the value of the game, we consider the difference

$$\left(\left(1/\left(p_{m+1}\ S\left(p\right)\right)\right) - \varepsilon\right)\ p_m - \frac{1}{S\left(p\right)} = \frac{p_m - p_{m-1}}{p_m\ S\left(p\right)} - \varepsilon p_m.$$

Since the first term on the right is positive because $p_m > p_{m-1}$, the difference will be positive for sufficiently small positive ε. $\square$

We will now consider an example to show how the solution changes as k goes up from the solved case of $k = n$. We conjecture that there exists a threshold with respect to k in which above that threshold, the hider 's optimal strategy is to hide at location n. To determine that threshold we use the following idea.

Proposition 5. *The game $G\left(n, p, t, k\right)$ has value $v = p_n$ if and only if the value v' of the game $G\left(n-1, \left(p_1, \ldots, p_{n-1}\right), \left(1, 2, \ldots, n-1\right), k-n\right)$ (with the last location removed and resources reduced by n) is at least p_n.*

Proof. Suppose $v = p_n$. Every search set with positive probability must include location n, otherwise simply hiding there implies $v < p_n$. So the remaining part of every search set has $k' = k - n$. With this amount of resources, the searcher must find the hider in the first n locations with probability at least p_n, which is what is stated in the Proposition. Otherwise, the searcher will either have to not search location n certainly (which gives $v < p_n$) or not search the remaining locations with enough resources to ensure $v \geq p_n$. $\square$

3.3.4 An example with $k = 10, n = 5$.

Consider the example where $p = (.5, .4, .3, .2, .1)$ with $k = 10, n = 5$. Here $p_n = .1$. The game with $p' = (.5, .4, .3, .2)$ and $k' = k - n = 5$ has value at least .1 because of the equiprobable search strategy of $\{1, 4\}$ and $\{2, 3\}$. Here each location in the new game is inspected with the same probability 1/2 and consequently the best the hider can do is to hide in the best location 4, and then the searcher wins with probability $(1/2)(.2) = .1$. It follows from Proposition 5 that the original game has the minimum possible value of $v = p_n = p_5 = .1$.

3.3.5 Illustrative examples

In this section we will use an example to further illustrate Proposition 4 and Corollary 8.

First, we consider the game where $k = n = 5$, $t_i = i$, and $p = (.5, .4, .3, .2, .1)$. The game matrix, excluding dominated search strategies, is given by

$A \backslash location$	1	2	3	4	5
$\{5\}$	0	0	0	0	.1
$\{1,4\}$	.5	0	0	.2	0
$\{2,3\}$	0	.4	.3	0	0
$\{1,3\}$	.5	0	.3	0	0
$\{1,2\}$	.5	.4	0	0	0

The unique solution for the optimal hiding distribution is $(0, 0, 2/11, 3/11, 6/11)$ and the value is $6/110 = 1/(1/.3 + 1/.2 + 1/.1) \simeq .055$. The optimal search strategy mentioned in Proposition 4 is to play $\{5\}, \{1,4\}$ and $\{2,3\}$ with respective probabilities 6/11, 3/11 and 2/11. Another strategy is to play $\{5\}$ and $\{1,4\}$ the same but to play $\{2,3\}$ and $\{1,3\}$ with probabilities 3/22 and 1/22. It is of interest to see how the solution of the game changes when k increases from $k = n = 5$ to higher values. We know that we need go no higher than $k = 10$ from Proposition 5 because in the game on locations 1 to 4 with $k' = 10 - 5 = 5$, the searcher can inspect $\{4,1\}$ with probability 2/3 and $\{3,2\}$ with probability 1/3 to ensure winning with probability at least $1/10 = p_5$.

So we know the solution of the game for $k = 5$ and $k \geq 10$. The following table 3.1 gives the value of the game and the unique optimal hiding distribution for these and intermediate values. (The optimal search strategies are varied and we do not list them, though they are easily calculated.)

$k \backslash i$	1	2	3	4	5	Value	
5	0	0	2/11	3/11	6/11	3/55	$\simeq 0.0545$
6	0	0	2/11	3/11	6/11	3/55	$\simeq 0.0545$
6	0	0	0	1/3	2/3	1/15	$\simeq 0.0667$
8	0	0	0	1/3	2/3	1/15	$\simeq 0.0667$
9	0	3/37	4/37	6/37	24/37	18/185	$\simeq 0.0943$
≥ 10	0	0	0	0	1	1/10	$= 0.1$

Table 3.1: Optimal hiding distribution and values, $k \geq 5$

We know that the value must be nondecreasing in k, but we see that it is not strictly increasing. Roughly speaking (but not precisely), the hider restricts towards fewer and better locations as k increases, staying always at the best location 5 for $k \geq 10$. However there is the anomalous distribution for $k = 9$ which includes sometime hiding at location 2.

3.3.6 Game with two types of locations

In this section we analyse a more specific scenario where all available hiding locations are of two types. This model might be vaguely applied to military practices. Suppose a team of law enforcement is to capture a hiding fugitive in an apartment complex. Then all possible hiding locations can be reduced to a number of types, e.g. smaller rooms have similar shorter search times and higher capture probability than a parking lot. Here we solve the resulting search-pursuit game.

Suppose there are two types of locations (hiding places). Type 1 takes time $t_1 = 1$ (this is a normalization) to search, while type 2 takes time $t_2 = \tau$ to search, with τ being an integer. Now let type 1 locations have capture probability p while type 2 locations have capture probability q. Moreover, suppose there are a locations of type 1 and b locations of type 2. The searcher has total search time k. To simplify our results we assume that k is small enough such that $a \geq k$ (the searcher can restrict all his searches to type 1) and $b\tau \geq k$ (he can also restrict all his searches to type 2 locations).

Let $m = \lfloor k/\tau \rfloor$ be the maximum number of type 2 locations that can be searched. The searcher's strategies are to search $j = 0, 1, \ldots, m$ type 2 locations (and hence $k - \tau j$ locations of type 1). Since all locations of a given type are essentially the same, the decision for the hider is simply the probability y to hide at a randomly chosen location of type 1 (and hence hide at a randomly chosen location of type 2 with probability $1 - y$).

Then the probability $P(j, y)$ that the searcher wins the game is given by

$$
\begin{aligned}
&yp(\frac{k-\tau j}{a}) - (1-y)q(\frac{j}{b}) \\
&= \frac{k}{a}py + \Big(\frac{q}{b}(1-y) - \frac{1}{a}py\tau\Big)j
\end{aligned}
$$

This will be independent of the searcher's strategy j if

$$
\begin{aligned}
&\frac{q}{b}(1-y) - \frac{1}{a}py\tau = 0, \text{or} \\
&\qquad\qquad y = \bar{y} \equiv \frac{aq}{aq + bp\tau}
\end{aligned}
$$

For $y = \bar{y}$, the capture probability is given by

$$
P(j, \bar{y}) = \frac{pqk}{aq + bp\tau}
$$

By playing $y = \bar{y}$, the hider ensures that the capture probability (payoff) does not exceed $P(j, \bar{y})$.

We now consider how to optimize the searcher's strategy. Suppose the searcher searches j locations of type 2 with probability x_j, $j = 0, 1, \ldots, m$. If the hider is at a type 2 location then he is captured with probability

$$
\begin{aligned}
&\sum_{j=0}^{m} x_j \frac{qj}{b} = \frac{q}{b} \sum_{j=0}^{m} j x_j = \frac{q}{b}\hat{\jmath}, \text{ where} \\
&\qquad \hat{\jmath} = \sum_{j=0}^{m} j x_j
\end{aligned}
$$

is the mean number of searches at type 2 locations. Similarly, if the hider is at a type 1 location, the hider is captured with probability

$$
\begin{aligned}
\sum_{j=0}^{m} x_j \frac{p(k-\tau j)}{a} &= \frac{pk}{a} - \frac{p\tau}{a}\sum_{j=0}^{m} j x_j \\
&= \frac{pk}{a} - \frac{p\tau}{a}\hat{\jmath}
\end{aligned}
$$

It follows that the capture probability will be the same for hiding at either location if we have

$$
\begin{aligned}
\frac{q}{b}\hat{\jmath} &= \frac{pk}{a} - \frac{p\tau}{a}\hat{\jmath}, \text{ or,} \\
\hat{\jmath} &= \frac{pbk}{bp\tau + aq}.
\end{aligned}
$$

So for any probability distribution over the pure strategies $j \in \{0, 1, \ldots, m\}$ with mean $\hat{\jmath}$, the probability of capturing a hider located either at a type 1 or a type 2 location is given by

$$
\frac{q}{b}\hat{\jmath} = \frac{pk}{a} - \frac{p\tau}{a}\hat{\jmath} = \frac{pqk}{aq + bp\tau}
$$

To summarize, we have shown the following.

Proposition 6. *Suppose all the hiding locations are of two types: a locations of type 1 with search time 1 and capture probability p; b locations of type 2 with search time τ and capture probability q. Suppose a and b are large enough so the searcher can do all his searching at a single location type, that is, $k \leq max(a, \tau b)$. Then the unique optimal strategy for the hider is to hide in a random type 1 location with probability $\bar{y} = \frac{aq}{aq+bp\tau}$ and in a random type 2 location with probability $1 - \bar{y}$. Note that this is independent of k. A strategy for the searcher which inspects j locations of type 2 (and thus, $k - j\tau$ for type 1) with probability x_j is optimal if and only if the mean number $\hat{\jmath} = \sum_{j=0}^{m} j x_j$, $m = \lfloor k/\tau \rfloor$ of type 2 locations inspected is given by $\hat{\jmath} = \frac{pbk}{bp\tau+aq}$. If this number is an* *integer, then the searcher has an optimal pure strategy. The value of the game is given by* $\frac{pqk}{aq+bp\tau}$.

3.4 Game Where Capture Probabilities are Unknown But Learned

In this section we determine how the players can *learn* the values of the capture probabilities over time, starting with some *a priori* values and increasing these at locations from which there have been successful escapes. This of course requires that the game is repeated. Here we consider the simplest model, just two rounds. So after a successful escape in the second round, we consider that the hider-prey has won the game (Payoff 0). More rounds of repeated play are considered in Gal, Alpern,Casas [11], but learning is not considered there.

We consider only two hiding locations, one of which may be searched in each of the two rounds. If the hider is found at location i, he is captured with a probability $1-q_i$ (escapes with complementary probability q_i). There are two rounds. If the hider is not found (searcher looks in the wrong location) in either round, he wins and the payoff is 0: If the hider is found and captured in either round, the searcher wins and the payoff is 1: If the Hider is found but escapes in the first round, the game is played one more time and both players remember which location the hider escaped from. If the hider escapes in the second (final) round, he wins and the payoff is 0.

The novel feature here is that the capture probabilities must be learned over time. At each location, the capture probability is chosen by Nature before the start of the game, independently with probability 1/2 of being h (high) and probability 1/2 and being l (the low probability), with $h > l$. In the biological scenario, this may be the general distribution of locations in a larger region in which it is easy or hard to escape from. A more general distribution is possible within our model, but this two point distribution is very easy to understand. If there is escape from location i in the first round, then in the second round the probability that the capture probability at i is h goes down (to some value less than 1/2). This is a type of Bayesian learning, which only takes place after an escape, and only at the location of the escape.

Our model contributes to the realistic interaction between searching-predator and hiding-prey acting in a possibly changing environment. Most often in nature, the searcher has no or incomplete information during the search and pursuit interaction. particularly in Mech, Smith, and MacNulty [53], a pack of wolves has to learn over time the difficulty of pursuing their prey in specific terrain. Moreover, hiding-prey such as elk seems to prefer areas with lots toppled dead trees, creating an entanglement of logs difficult to travel through. We focus here on asking questions if learning the capture probabilities will affect the searching and hiding behaviour. More specifically, suppose an elk manages to escape through the deep forest, should it stay there where he believes the capture probability is low enough, or hide at a different location?

3.4.1 Normal form of the two-period learning game

We use the *normal form* approach, rather than a repeated game approach. A strategy for either player says where he will search/hide in the two periods (assuming the game goes to the second period). Due to the symmetry of the two locations, both players cannot but choose their first period search or hide locations randomly. Thus the players have two strategies: rs (random,same) and rd (random,different). If there is a successful escape from that location, they can either locate in the same location (strategy rs) or the other location (strategy rd). This gives a simple two by two matrix game. In this subsection we calculate its normal form; in the next subsection we present the game solution.

First we compute the payoff for the strategy pair (rs, rs): Half the time both players (searcher and hider) go to different locations in first period, in which case the hider wins and the payoff is 0. So we ignore this, put in a factor of (1/2), and assume they go to the same location in the first period. There is only one location to consider, suppose it has escape probability x. Then, as they both go back to this location in the second period if the hider escapes in the first period, the expected payoff is given by

$$P_x(rs, rs) = (1/2) \ \ ((1-x)1 + x(1-x)). \tag{3.1}$$

Since x takes values l and h equiprobably we have

$$\begin{aligned} P(rs, rs) &= \frac{P_h(rs, rs) + P_l(rs, rs)}{2} \\ &= \frac{2 - h^2 - l^2}{4}. \end{aligned} \tag{3.2}$$

It is worth noting two special cases: If both escape probabilities are 1 (escape is certain), then the hider always wins and the payoff is 0. If both escape probabilities are 0 then the searcher wins if and only if they both choose the same location, which has probability 1/2.

Next we consider the strategy pair (rd, rd). Here we can assume they both go to location 1 in the first period (hence we add the factor of 1/2) and location 2 in the second period. The escape probabilities at these ordered locations 1 and 2 can be any of the following: hh, ll, hl, lh. The first two are straightforward as it is the same as going to the same location twice (already calculated in (3.2)). We list the calculation of the four ordered hiding locations below, where P_x is given in (3.1).

$$P_{hh}(rd, rd) = P_h(rs, rs)$$

$$P_{ll}(rd, rd) = P_l(rs, rs)$$

$$P_{lh}(rd, rd) = (1/2)((1-l)1 + l(1-h))$$

$$P_{hl}(rd, rd) = (1/2)((1-h)1 + h(1-l))$$

Taking the average of these four values gives,

$$P(rd, rd) = \frac{4 - h^2 - l^2 - 2hl}{8} = \frac{4 - (h+l)^2}{8}. \tag{3.3}$$

Now consider the strategy pair (rs, rd). If they go to different locations in the first period, the game ends with payoff 0. So again, we put in factor of 1/2 and assume they go to same location in first period. This means that if an escape happens in the first period, the hider wins (payoff 0) in the second period. So the probability the searcher wins is

$$\begin{aligned} P(rs, rd) = P(rd, rs) &= (1/2)\Big(1/2((1-h) + (1-l))\Big) \\ &= \frac{2 - (h+l)}{4}. \end{aligned} \tag{3.4}$$

Thus, we have completed the necessary calculations and the game matrix for for the strategy pairs rs and rd, with searcher as the maximizer.

To solve this game, we begin with the game matrix as follows;

$$A = A(l,h) = \begin{bmatrix} P(rs,rs) & P(rs,rd) \\ P(rd,rs) & P(rd,rd) \end{bmatrix}$$
$$= \begin{bmatrix} \frac{2-(h^2+l^2)}{4} & \frac{2-(h+l)}{4} \\ \frac{2-(h+l)}{4} & \frac{4-(h+l)^2}{8} \end{bmatrix}$$

Then we take out the fraction 1/8 to the left hand side of the equation, and we have

$$8A = \begin{bmatrix} -2h^2 - 2l^2 + 4 & 4 - 2h - 2l \\ 4 - 2h - 2l & 4 - (h+l)^2 \end{bmatrix}$$

At this point we try to make the right-hand side of the equation to be a diagonal matrix so we can easily compute it. Therefore we can write the equation as follow

$$8A - (4 - 2h - 2l) \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = Y = \begin{bmatrix} -2h^2 + 2h - 2l^2 + 2l & 0 \\ 0 & 2h + 2l - (h+l)^2 \end{bmatrix}.$$

Note that $V(A)$ is the value of the matrix A. From the equation above, it shows that the right-hand side of the equation is a diagonal matrix, and a simple formula for the value of diagonal matrix games is as follow

$$V\left(\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}\right) = 1/\left(1/a + 1/b\right).$$

Using the above formula, we have

$$V\left(8A-(4-2h-2l)\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}\right)=V\left(Y\right)=\frac{1}{\frac{1}{-2h^2+2h-2l^2+2l}+\frac{1}{2h+2l-(h+l)^2}}.$$

Computing this for the value of game matrix A, we have the following equation for $V(A)$,

$$V\left(A\right)=\frac{1}{2}-\frac{1}{4}l-\frac{1}{4}h-\frac{1}{8\left(\frac{1}{2h^2-2h+2l^2-2l}-\frac{1}{2h+2l-(h+l)^2}\right)}. \tag{3.5}$$

It is also important to note that in a diagonal game, players adopt each strategy with a probability inversely proportional to its diagonal element. To obtain this we first calculate the value of $V(Y)$ given above. Then, both the searcher and hider should choose rs and rd with probabilities $V(Y)/a$ and $V(Y)/b$ respectively.

We can now see that, as expected, a successful escape from a location makes that location more attractive to the hider as a future hiding place. This is confirmed in the following.

Proposition 7. *In the learning game when $l < h$, after a successful escape both players should go back to the same location with probability greater than 1/2.*

Proof. Let a and b denote, as above, the diagonal elements of Y. We have

$$\begin{aligned} a-b &= \left(-2h^2+2h-2l^2+2l\right)-\left(2h+2l-(h+l)^2\right)\\ &= -\left(h-l\right)^2<0. \end{aligned}$$

This means that $b > a$ and $V/a > V/b$. Hence by observation (3.5) the strategy rs should be played with a higher probability (V/a) than rd (probability V/b), in particular with probability more than 1/2. □

3.4.2 An example with $l = 1/3$ and $h = 2/3$

A simple example is when the low escape probability is $l = 1/3$ and the high escape probability is $h = 2/3$: This give the matrix A as

$$A(l,h) = \begin{bmatrix} \frac{13}{36} & \frac{1}{4} \\ \frac{1}{4} & \frac{3}{8} \end{bmatrix}$$

with value $V = V(1/3, 2/3) = 21/68$, and where each of the player optimally plays rs with probability 9/17 and rd with probability 8/17.

Suppose there is an escape in the first period at say location 1. Then in the second period the hider goes to location 1 with probability 9/17. Since the subjective probability of capture at location 2, from the point of view of either player, remains unchanged at (1/3 + 2/3) /2 = 1/2; this corresponds to a certain probability x at location 1, that is, a matrix

$$\begin{bmatrix} x & 0 \\ 0 & 1/2 \end{bmatrix}$$

We then have that

$$(9/17)x = (8/17)(1/2) \text{ or,}$$
$$x = 4/9.$$

This corresponds to the probability of escape probability $l = 1/3$ of q, where

$$q1/3 + (1-q)2/3 = 4/9 \text{ or,}$$
$$q = 2/3.$$

Thus, based on the escape at location 1 in the first period, the probability that the escape probability there is 1/3 has gone up from the initial value of 1/2 to the higher value of 2/3.

3.5 Conclusion

The breakthrough paper of Gal and Casas [36] gave us a model in which both the search and pursuit elements of predator-prey interactions could be modeled together in a single game. In that paper the capture probabilities depended on the hiding location but the time required to search a location was assumed to be constant. In the first part of this paper, we drop that simplifying assumption. We first consider a particular scenario where we order the locations such that the search times increase while the capture probabilities decrease. We solve this game for the case of a particular total search time of the searcher. We then consider a scenario where there are many hiding locations but they come in only two types. Locations of each type are identical in that they have the same search times and the same capture probabilities. We solve the resulting search-pursuit game.

In the second part of the paper we deal with the question of how the players (searcher-predator and hider-prey) learn the capture probabilities of the different locations over time. We adopt a simple Bayesian approach. After a successful escape from a given location, both players update their subjective probabilities that it is a location with low or high capture probability; the probability that it is low obviously increases. In the game formulation, the players incorporate into their plan the knowledge that if there is an escape, then that location becomes more favorable to the hider in the next period.

The search-hide and pursuit-evasion game is quite difficult and finding a solution for the most general case is quite challenging. Most probably, it is a good idea for the next step to solve for a more specific question in the problem.

We consider a possible extension to Proposition 6 by analysing larger k. Consider the example $a = b = 1; t_1 = 1; t_2 = 3; k = 4$; and say $p < q$ ($p_1 < p_2$ as in Gal- Casas [36]). The Searcher inspects both cells (one of each type), so he certainly finds the Hider. He captures him with probability p if the hider is at location 1, and q if at location type 2. So the Hider should hide at location of type 2 as it has lower capture probability. The main question will be: How big does k have to be for this to occur? And are there only two solution types ? We conjecture that, as in Gal-Casas, there is a critical value of $k = \hat{k}$ such that for $k < \hat{k}$, Proposition 6 applies, and for $k \geq \hat{k}$ the Hider locates in a cell of the type with the lower capture probability.

The game with learning model has also been analysed using dynamic form in [12]. This allows more effective analysis for more than two locations and two periods. Moreover, we believe the next avenue of research is to consider the non-zero-sum game. Indeed, one may argue that a game between a predator and a prey may not necessarily be a zero-sum game, as the predator is hunting its dinner while the prey is running for survival. This is an important if challenging aspect to deal with for future studies.

Chapter 4

Optimizing Voting Order on Sequential Juries: A Sealed Card Model

4.1 Introduction

Jury voting between equiprobable binary alternatives was first studied by Condorcet [29]. In his model, each of an odd number of n jurors receives an independent signal as to which alternative holds (say, "innocent" or "guilty"), and they vote according to their signal. The signals are only correct with a common probability $p > 1/2$. The so-called Condorcet Jury Theorem says that the reliability (i.e., the probability that the majority verdict is correct) goes to 1 as n goes to infinity. Condorcet's analysis has been extended in many directions. The direction that concerns us here is to make the voting sequential, so that when a juror votes he is aware of all previous votes. With sequential voting, another issue appears. If the jurors have different "abilities" (say different values of p in the Condorcet model), which voting order is most likely to give the correct majority verdict (has the highest *reliability*)? Alpern and Chen [2] introduced a numerical model in which abilities are indicated by a discrete set of five equally spaced abilities $\{0, 1, 2, 3, 4\}$ and where signals lie in the interval $[-9, -7, \ldots, 7, 9]$, with the extremes giving maximum probability for each alternative. In that model, they found that given a jury of three fixed abilities, the optimal voting order is as follows: the juror of middle ability votes first, then the juror of the highest ability, followed by the juror of the lowest ability. For any jury of three, and any notion of ability, we shall call this the *Alpern-Chen ordering.*

In Alpern and Chen [2] the notion of ability was abstract and defined as a parameter in the signal distribution. Here we have a more concrete notion of ability — it is an integer giving the number of cards a juror is allowed to sample in order to guess the colour of the sealed card. It is the main motivation of the present paper to give an appealing concrete example of the notion of ability. Specifically, the model is as follows. A card is removed ("sealed") unseen from a deck of size $D = 2m$, half red (R) and half black (B). The colour of the sealed card is the state of Nature (N), which the jurors attempt to guess in their vote. Each juror is allowed to view a sample of cards, where the sample size is his ability. He also has access to the information about the votes of jurors voting earlier together with their "abilities" (i.e., sample sizes). Each juror is constrained to vote for the alternative (R or B) that is more likely, conditioned on his private information (the number of red cards in his sample) and the prior voting. Given a set of jurors of known abilities, in which order should we let them vote to maximize the probability that the majority verdict is correct?

More generally, the main aim of this paper is to compare the efficacy of the six voting orders for a jury of three abilities $a < b < c$. We do this for various deck sizes D. We consider three criteria: The first is the *reliability*, which depends on the voting order and the three abilities and gives the probability that the majority verdict of such a jury is correct. The second is the *optimality fraction* of a voting order, which is the fraction of all ability sets for which it has the highest reliability. The third criteria is the *average reliability* of a voting order, which averages the reliabilities over all possible sets of abilities. We observe that if for some deck size D a certain voting order (say the Alpern-Chen order) has the highest reliability for every ability set, then it also has the highest average reliability and has an optimality fraction of 1. This will be more formally defined in Section 4.3.

To conclude this section, we give the reader an illustrative example showing the importance of voting order on the reliability of the majority verdict.

Example 1. *Suppose the original deck has $D = 4$ cards, two red and two black. Suppose there are three jurors, and juror $i = 1, 2, 3$ can sample i cards. Note that juror 3 will know the color of the sealed card, so if he votes first or second the later voting jurors will copy him to produce an accurate majority verdict with reliability 1. So we need only consider the two orderings where juror 3 votes last. It is easiest to calculate the probability that the verdict is wrong. We may assume that the sealed card N is R, by symmetry. We calculate the probability of initial voting $BB\_$: First consider the voting order $(1, 2, 3)$. Juror 1 votes B if he draws a red card, which has probability $1/3$. The only case where juror 2 will not copy is if he draws two black cards, which has probability 1/3. So the probability of $BB\_$ is $(1/3)(2/3) = 2/9$; and the reliability is $1 - 2/9 = 7/9$. Next, consider the voting order $(2, 1, 3)$. Juror 1 will vote red unless he picks a red and black, in which case he randomizes. So he votes B with probability $(2/3)(1/2) = 1/3$. If juror 2 draws a red, he also votes B. If he draws a black, he votes R. The probability he draws a red is 1/3. so the probability of $BB\_$ is $(1/3)(1/3) = 1/9$ and hence the reliability of jury $(2, 1, 3)$ is $1 - 1/9 = 8/9$.*
Thus we have shown the reliability is 1 if juror 3 votes last, is $7/9$ for the voting order $(1, 2, 3)$ and is $8/9$ for $(2, 1, 3)$.

Example 1 shows the importance of voting order. Note in particular that the Alpern-Chen order $(2, 3, 1)$ is among the optimal orderings. The difficulty in obtaining this type of result on optimal voting order was highlighted in the well-known paper on sequential juries of Dekel and Piccione [31]. We also note that our model is distinct from that of Alpern-Chen, so it was not clear from the beginning that their optimal ordering would also be optimal in our model. Indeed for some juries it is not. For example when $D = 6$ with three red and three black, and abilities $i = 1, 2, 3$. We found that voting order $(1, 3, 2)$ is the unique optimal ordering. However, the fact that we get the same optimal ordering in a different probability model suggests that the Alpern-Chen ordering could heuristically be utilised by decision-maker to achieve a reliable final decision via sequential voting.

Summary of the main results

The main results of this paper are propositions based on computations from a finite number of cases with Mathematica and MatLab, and are reported in Sections 4.4 – 4.6. Most of these results are quantitative in nature, for example, we give precise probability of the three jurors reaching the correct verdict given the deck size, their ability, and the voting order. However, we believe that it is useful to give rough summary of some of these results here. For the precise results on which these summaries are based, please refer to the specific propositions quoted.

1. **For small decks (of size $D \leq 16$):** The Alpern-Chen ordering maximizes the reliability of the majority verdict for any jury with even abilities. The seniority ordering (highest, middle, lowest) has higher reliability than the anti-seniority ordering (lowest, middle, highest). Whichever juror votes first, the reliability is higher when the next two jurors vote in decreasing order of ability. See Propositions 8, 9, and 10.
2. **For medium size decks ($D \leq 52$):** For each D, the Alpern-Chen ordering is best for more juries than any other ordering (Proposition 11). For each D, the average reliability of the Alpern-Chen ordering, taken over all juries, is higher than for any

other ordering (Proposition 12).

3. **Comparison of the reliability of sequential scheme (in the Alpern-Chen order) with the more familiar simultaneous voting scheme**: For a very large jury, sequential voting eventually herds, so that unlike the Condorcet Jury Theorem, the reliability does not tend to 1. We give a comparative analytical argument (Theorem 9) that simultaneous voting has higher reliability than sequential for the homogeneous jury with common ability 1 and $D \geq 4$, while this is reversed for common ability 2 and $D = 4$. We show that in general sequential voting is better than simultaneous in terms of average reliability (Proposition 14) and in terms of the fraction of juries for which it is better (Proposition 15). We then ask the question of when (even if rarely) simultaneous voting is better. The answer is for relatively homogeneous juries (ones with low variability of ability).

4. **For a larger jury of size 5**: We check to what extent our main results of juries of size 3 carry over. We find that when D is between 12 and 30, the five voting orders of highest reliability (out of 120 possible orderings) all have the most able juror votes in the middle of the order (as in the Alpern-Chen ordering). We find other analogies as well.

4.2 Related literature

The founding paper of this literature is Condorcet [29]. The two directions in which we travel from that source are heterogeneous jurors (differing abilities) and, more importantly, sequential voting. Heterogeneous jurors, in the sense of differing correctness probabilities p in the Condorcet sense, have been considered in many papers, notably Hoeffding [40], Boland [22] and Kanazawa [44]. Sequential voting has been much less studied. Two significant papers in this direction are Dekel and Piccione [31], which compares simultaneous and sequential equilibria, and Ottaviani and Sørensen [57], which considers honest voters concerned with their reputations.

Dekel and Piccione [31] discuss the comparison between simultaneous and sequential voting mechanisms. They show that the symmetric equilibria in the simultaneous voting game are also equilibria of the sequential voting. Moreover, they examine some instances where information aggregation in sequential voting game does no better than one found in the simultaneous scheme. Our finding provides an interesting contrast to this finding as we find in the sealed card problem that sequential voting most often results in a better voting decision compared to the simultaneous voting mechanism.

Ottaviani and Sørensen [57] consider a jury of heterogeneous experts who care only about their own vote being correct, to enhance their future reputation. They keep the assumption of binary signals. They also emphasize the importance of group thinking and herding behaviour.

The paper that introduces a model concerned with the optimal ordering of voters with heterogeneous abilities is Alpern and Chen [2]. Each of three jurors receives a signal (private information) in the interval $[-9, -7, \ldots, 7, 9]$. The distribution of the signal depends on the binary state of Nature and the ability of the juror. From the signal and previous votes, the juror determines the probability of the two states of Nature and votes for the more likely one. The middle-high-low ordering is shown to be optimal in terms of the reliability of the majority verdict. That paper also considered strategic voting, where jurors can use voting rules that maximize the reliability of the group verdict, rather than the probability that their own vote is correct. We will not consider strategic voting in this paper.

In the situation where the hierarchy or the experience of the jurors has high variability, conventional wisdom holds that the voting order follows the seniority rule of decreasing ability order [43], as for most of the cases in the US Supreme Court. Moreover, when the case is uncertain, or a tie occurs in a vote, then the most senior judge should cast the casting vote. This seniority rule is in an interesting contrast to the work of Alpern and Chen [3] which establishes that voting achieves the highest reliability when the casting vote is allocated to the juror with median ability.

Another relevant paper is by Kanazawa [44]. He shows that as long as the mean of individual competence is high, then heterogeneous groups are better at making the correct decision than homogeneous groups for any given level of mean competence. The significance of voting order in a committee version of the Secretary Problem is analyzed by Alpern, Gal and Solan [13] and Alpern and Baston [1] in a model that allows vetoes.

4.3 The Sealed Card Voting Model

In this section, we formally model the sealed card problem. The parameters of the model are the size n of the jury, taken as odd so that a majority always exists, and the size $D = 2m$ of the deck of cards, m are red (R) and m are black (B). Also, the ability $a_i \in \{1, \ldots, D-1\}$ of each juror i is known. The model is dynamic: the first step is a random selection of a card, which is removed unseen (*sealed*) from the deck. The colour of the sealed card, R or B, is the state of Nature ($N \in \{R, B\}$) that the jurors will attempt to guess correctly. Then each juror who votes in position $i \in \{1, \ldots, n\}$ is allowed to see a_i cards drawn randomly from the remaining deck of $D-1$ cards, which we will call *card pool* from now on. We refer back to Example 1 and assume that the sealed card is red. Having said so, we know there are $m-1$ red cards and m black cards in the card pool. In general, given a juror of ability a, we can write the probability of randomly drawing x red cards out of a total $m-1$ red cards in the card pool as follows:

$$\Pr[x|m-1, a] = \frac{\binom{m-1}{x}\binom{m}{a-x}}{\binom{2m-1}{a}}. \tag{4.1}$$

Based on the number of red and black cards in his sample, and the prior voting, juror i uses Bayes' Law to decide the most likely state of Nature. If R and B are equally likely, the juror votes randomly. He then votes this way (R or B) and returns his sample to the card pool. After the last juror votes, the verdict of the jury is the majority colour, denoted by V, in the odd voting sequence. We (or the organizer of the jury) wish to compare the voting order of the jurors of given abilities in order to maximize the probability (which we called the *reliability*) that the verdict $V \in \{R, B\}$ is correct. For example, if $n = 3$ and the abilities are distinct, say 1, 2, 3 there are $3! = 6$ ability orders to consider. For a fixed set of abilities (like $\{1, 2, 3\}$ in Example 1), each voting order has a reliability that can be calculated and compared to the others.

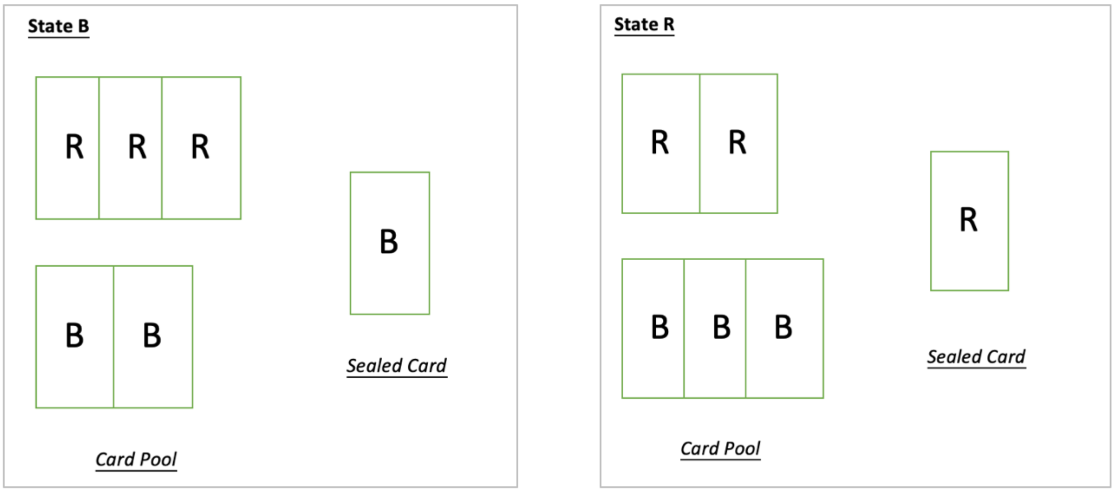

Figure 4.1: The two states of Nature, $N = B$ (left), $N = R$ (right), $D = 6$

Consider for example a deck of $D = 6$ cards, with the two states of Nature illustrated in Figure 4.1. If juror $i = 1$ with ability $a_1 = 1$ draws a red card, he knows he is more likely in State B and consequently votes B. Suppose juror $i = 2$ with ability $a_2 = 4$ draws a sample of two red and two black cards. From his private information alone, he views the two states as equiprobable and, if voting first, would vote randomly. However, if he knows that the first juror voted B with ability of $a_1 = 1$, this increases the conditional probability that Nature is B to above one half, he would vote B as well. This is a type of herding. Of course, if the second juror drew three black cards, he would vote R.

Voting behaviour. We assume that every juror votes honestly for the alternative, R or B, that he views as more likely, given prior voting with knowledge of prior abilities and the cards he draws. If considering our model as a game, this would mean each juror gets a payoff of 1 if his vote turns out to be correct and 0 otherwise. He does not care about the correctness of the verdict. This type of honest voting would likely prevail if the juror is rewarded, perhaps in improving their reputation, if his vote (rather than the majority verdict) turns out to be correct. If after sampling he still views the state of Nature as equally likely, then he randomizes his vote.

Abilities. Denote by $\Lambda = \{1, 2, \ldots, D-1\}$ the set of juror *abilities*, where $D-1$ is the size of the card pool. Saying that a juror has ability a means that when he comes to vote, he samples a cards from the card pool. An *ability set* $A \subseteq \Lambda$ is a set of n distinct abilities, considered as an *unordered jury.* Thus the set $\mathcal{A}_n$ of all unordered juries of size n is as follows:

$$\mathcal{A}_n = \{A \subseteq \Lambda : |A| = n\},$$

where we use $|A|$ to denote the cardinality of set A. We list the abilities of an unordered jury in increasing order for convenience, such as $A = \{4, 5, 6\}$. Let Π_n denote the set of all $n!$ rank orderings (permutations) of $\{1, \ldots, n\}$. A *jury* J of size n is an *ordered* set of n distinct abilities. Let $\mathcal{J}_n$ denotes the set of all juries of size n. There is a natural way to apply an ordering $r \in \Pi_n$ to an ability set $A \in \mathcal{A}_n$ to obtain a jury J. For example, if $n = 3$, $r = (2, 1, 3)$ and $A = \{5, 7, 9\}$, then we can write $r(A) = (7, 5, 9)$. This means the ordered jury $(7, 5, 9)$ has the first juror of ability 7 votes first, followed by the second juror of ability 5, and the last juror to vote is the one with ability 9. In other words, the elements of the set A are written in the rank order of r. Thus we can think of each rank order r as a mapping of ability sets into juries, $r : \mathcal{A}_n \to \mathcal{J}_n$.

Reliabilities and other measures. For every jury $J \in \mathcal{J}_n$, we define its *reliability* $Q(J) \in [0, 1]$ as the probability the jury gets the verdict correctly, that is, the majority verdict V is the same as the colour N of the sealed card. . The reason for the reliability in probabilistic form is the stochastic nature of the sampling. Thus, $Q : \mathcal{J}_n \to [0, 1]$ is a mapping from juries to probabilities. The problem we address is how to optimize the voting order given an ability set.

For every rank ordering $r^* \in \Pi_n$, we define its *optimality fraction* $\phi(r^*)$ as the fraction of ability sets $A \in \mathcal{A}_n$ for which it gives the highest reliability, that is, for which $Q(r^*(A)) \geq Q(r(A))$ for all $r \in \Pi_n$. Note that this function ϕ is dependent upon the deck size. More precisely,

$$\phi(r^*) = \frac{|\{A \in \mathcal{A}_n : Q(r^*(A)) \geq Q(r(A)) \text{ for all } r \in \Pi_n\}|}{|\mathcal{A}_n|}. \tag{4.2}$$

For $r \in \Pi_n$, we also define its *average reliability* $\bar{Q}(r)$ as a measure of how reliable this voting order is for a random set of abilities. More precisely, $\bar{Q}(r)$ is the average value of $Q(r(A))$ for ability sets $A \in \mathcal{A}_n$. Thus, $\bar{Q} : \Pi_n \to [0, 1]$ is a mapping from rank orderings to probabilities.

In the remainder of this section, we describe specifically the process of private information amalgamation. We begin in Section 4.3.1 by discussing the *public bias* θ, *voting threshold*, and the existence of *herding*. Then in Section 4.3.2, we present the formula for calculating the reliability.

4.3.1 Public bias, voting threshold and herding

When a juror comes to vote, before he looks at his sample of cards, there is a probability θ, conditional on prior voting, that the sealed card is red. We call this probability the *public bias*. We call it public because it is conditioned only on public information (voting) and not private information (cards drawn). For the first juror, the public bias is given by $\theta = 1/2$, or $\theta_1 = 1/2$ to indicate it is the public bias of the first juror. It is $1/2$ because as the original deck has the same number of red and black cards, the randomly chosen sealed card has probability one half of being red. If juror i votes R, the bias θ_{i+1}

goes up (from θ_i) for the next juror $i+1$; if he votes B, it goes down. The change from θ_i to θ_{i+1} does not depend directly on the cards drawn by juror i, as this is not known by juror $i+1$. The precise way in which θ changes after a vote is determined by Bayes' Law. We note that the public bias θ_{i+1} could be calculated by anyone (not necessarily a juror) who knows the abilities of the first i jurors and how they voted.

Next, suppose a juror of ability a comes to vote when the public bias is θ. Depending on the number x of red cards in his sample of a cards, the probability of $N = R$ (sealed card red) is either $> 1/2$, $< 1/2$ or in rare cases $= 1/2$. His corresponding vote will be to vote B, R, or randomize equiprobably. Before looking at his sample of cards he can compute a threshold τ such that if $x > \tau$ the probability of B is $> 1/2$; if $x < \tau$ the probability of B is $< 1/2$; if $x = \tau$ the probability of B is $1/2$. His corresponding votes will be B, R, or randomize. Note that say $\tau = 3.1$ and $\tau = 3.2$ are equivalent thresholds in that they produce the same votes. For convenience we can take $\tau = 3.5$ in such cases for uniqueness.

So generally $\tau = (k+1)/2$ for some non-negative integer k. However, if say $x = 4$ results in equiprobable R and B, and thus randomized voting, we say that $\tau = 4$. If $\tau = a+1/2$, with a denoting his ability, the juror will certainly vote R because he cannot have x more than a In such a case ($\tau > a$) we say that there is herding – the juror can vote without even looking at his cards. Similarly if $\tau < 0$, the juror certainly votes B. In our model, herding may be temporary: one juror might herd while the next juror (with a higher ability) might not. If we allowed jurors with $a = 0$ ability, they would always herd. In general, herding is less likely with jurors of higher ability. If the jurors vote in seniority order (decreasing order of ability) then once one juror herds, all subsequent jurors also herd.

4.3.2 Calculation of reliabilities

In this section, we derive formulas for reliability of a jury of size $n = 3$. Suppose Nature is in state R. Then the three voting sequences $(R, R,\ \text{any})$, (R, B, R) and (B, R, R) lead to a correct verdict. Note that if the first two jurors vote for the same colour, the verdict has already been generated by majority rule, no matter which colour the third juror votes. Similarly, the three voting sequences (B, B, any), (B, R, B) and (R, B, B) lead to the correct verdict if $N = B$. Having stated this, we ask the simple question: Given a deck of size $D = 2m$ (≤ 52) and three jurors of abilities a, b, c, in what order should the three jurors vote to amalgamate their private information best and so produce the highest voting reliability?

We will show that the reliability is maximized when the three jurors vote in Alpern-Chen ordering, under two kinds of measurement: optimality fraction and average reliability. For every jury $J = (a, b, c)$ of public bias of θ, the reliability $Q(J)$ is calculated as follows:

$$
\begin{aligned}
Q(a, b, c) = \theta_1 (&\Pr[RR\_] + \Pr[RBR] + \Pr[BRR]) \\
&+ (1 - \theta_1)(\Pr[BB\_] + \Pr[BRB] + \Pr[RBB]).
\end{aligned}
$$

The equation above computes the probability that the majority verdict is R when $N = R$ or the probability that the majority verdict is B when $N = B$. For example, $\Pr[RR\_]$ is the probability of the first and the second juror both votes Red. Note that the third voter in $\Pr[RR\_]$ is blank because the majority vote is already decided when the first two jurors voted R. In the beginning of Section 4.3.1 we stated that $\theta_1 = 1/2$ because the original deck has the same number of red and black cards. This also means that the states $N = R$ and $N = B$ are symmetric. Therefore, the equation above can be updated as follows:

$$\begin{aligned} Q(a,b,c) &= 1/2\ (\Pr[RR\_] + \Pr[RBR] + \Pr[BRR]) \\ &\qquad + 1/2\ (\Pr[BB\_] + \Pr[BRB] + \Pr[RBB]). \\ &= \ (\Pr[RR\_] + \Pr[RBR] + \Pr[BRR]). \end{aligned} \tag{4.3}$$

To compute the reliability (4.3) above, we use the hypergeometric probability represented in (4.1) and its cumulative distribution function (CDF).

$$\Pr[Y \le \tau \mid a] = \sum_{i=0}^{\tau} \frac{\binom{m-1}{i}\binom{m}{a-i}}{\binom{2m-1}{a}}. \tag{4.4}$$

Suppose the sealed card is red (i.e., $N = R$) so there are $m-1$ red cards and m black cards in the card pool. The equation (4.4) computes the probability of drawing up to Y red cards from the possible $m-1$ red cards in the card pool in a drawings without replacement from a deck of $2m-1$. The probability of $Y \le \tau$ is the cumulative distribution. Then for a jury $J = (a,b,c)$, the probabilities of voting sequences in (4.3) are now calculated in the following formulae, where $\tau(\sigma)$ denotes the threshold of juror who has the prior voting sequence σ:

$$\begin{aligned} \Pr[RR\_] &= \Pr[Y \le \tau \mid a]\ \Pr[Y \le \tau(R) \mid b], \\ \Pr[RBR] &= \Pr[Y \le \tau \mid a]\ (1 - \Pr[Y \le \tau(R) \mid b])\ \Pr[Y \le \tau(RB) \mid c], \\ \Pr[BRR] &= (1 - \Pr[Y \le \tau \mid a])\ \Pr[Y \le \tau(B) \mid b]\ \Pr[Y \le \tau(BR) \mid c]. \end{aligned}$$

To check this, let us refer back to Example 1 and calculate $Q(1,2,3)$ using the formulae above. We know that the first juror with ability $a = 1$ will vote R if he draws 0 red card, thus $\Pr[Y \le 0 \mid 1] = 2/3$. If juror 1 voted R, juror 2 with ability $b = 2$ will vote R with $\Pr[Y \le 1 \mid 2] = 1$ and vote B with probability 0. However if juror 1 voted B, juror 2 will vote R with probability 1/3. Juror 3 with ability $c = 3$ will always vote R

because his ability is equal to the card pool, so he is able to draw all the cards. Hence, $Q(1,2,3) = (\Pr[RR\_] + \Pr[RBR] + \Pr[BRR]) = 2/3*1 + 2/3*0*1 + 1/3*1/3*1 = 7/9$. This result is the same as the one we calculated in Example 1.

4.4 Optimal Voting Orders For $n = 3$

We are now ready to present our findings on optimal voting order for a jury of size three. We begin in Section 4.4.1 by studying a smaller deck and limiting each juror to even-only ability. In Section 4.4.2, we present our result on the optimality fractions ϕ. Section 4.4.3 considers optimal voting orderings in terms of the average reliability $\bar{Q}$. Apart from Section 4.4.1, the rest of our paper analyses the normal 52-card deck and all ability (odd and even) sets.

4.4.1 On reliability for small decks

In this subsection, we show that when the deck size D satisfies $D \leq 16$ and all jurors have even abilities, then the Alpern-Chen ordering results in highest reliability. We have also tested this with odd ability and arbitrary (both odd and even) ability in which we did not obtain a clear result with respect to reliability. We suspect that there are two reasons for this. First, note that if the first juror has even ability, this allows for ties (the state of Nature is still equally likely even after sampling), for which the juror randomizes his vote. The resulting random voting tends to smooth out certain elements of the reliability function and gives more consistent patterns of optimal reliability. Secondly, when all jurors have odd ability, then this allows for complete information regarding the true State of Nature. As an example, for deck size $D = 8$, and we have an odd-only ability set $\{3, 5, 7\}$, whenever a juror of ability 7 votes first or second, the reliability will be equal to 1. We are not able to see a clear "ranking" of the voting orderings in terms of reliability as opposed to even-only ability sets. The computed data establishing these propositions is given in Table 4.6 in the Appendix.

Proposition 8. *For a three-member jury of even-only abilities and deck size $D \leq 16$, the Alpern-Chen ordering (median-high-low) gives the strictly highest reliability.*

Many discussions in the literature of optimal voting orders compare only seniority and anti-seniority ordering. For small decks we can give a definitive answer as to which is better.

Proposition 9. *For three-member jury of even-only abilities and deck size $D \leq 16$, the seniority ordering (c, b, a) has the higher reliability than the anti-seniority ordering (a, b, c), where $a < b < c$.*

Proposition 10. *For deck size $D \leq 16$ and a three member jury of even abilities x, a, b with $a < b$, the ordering (x, b, a) has a higher reliability than (x, a, b). That is, it is always better for the last to jurors to vote in decreasing order of ability.*

Note that Proposition 10 is applicable in the situation where the first voter is already pre-determined, such as most often in a company, the moderator (not necessarily of highest ability) gets to voice the initial opinion. Then the lower ability between the remaining two should always vote last. The readers are encouraged to see Table 4.6 in the Appendix for our computational 'proof' of Proposition 8, 9, and 10.

4.4.2 On optimality fraction

In this subsection we calculate the optimality fraction for the six orderings of a three-member jury. Recall that this is the fraction of the potential ability sets for which the ordering gives the highest reliability. In most of our discussion below, we use the case of $D = 6$ for explanatory reason. Then we give a result (Proposition 11) that holds for $D \leq 52$.

Table 4.1 summarizes our findings about the performance of different voting orderings in terms of their optimality fraction ϕ for decks of sizes range from 6 to 52 (for

Optimality fraction ϕ						
Deck size / Voting Orders	$D = 6$	8	...	26	...	52
(a, b, c)	0/10=0.0	0.029	...	0.006	...	0.007
(a, c, b)	8/10=0.8	0.620	...	0.298	...	0.200
(b, a, c)	0/10=0.0	0.000	...	0.000	...	0.000
(b, c, a)	**9/10=0.9**	**0.800**	...	**0.652**	...	**0.630**
(c, a, b)	6/10=0.6	0.429	...	0.167	...	0.148
(c, b, a)	6/10=0.6	0.486	...	0.305	...	0.253
Sum	2.9	2.371	...	1.427	...	1.238

Table 4.1: Optimality fraction ϕ for deck size up to 52, $(a < b < c)$

fuller results the reader is referred to Table 4.7 in the Appendix). To get a small taste of our findings, observe Table 4.2, which corresponds to the case of deck size $D = 6$. Note that for $D = 6$, we have the following $\binom{6-1}{3} = 10$ ability sets:

$$\mathcal{A}_3 = \{\{1,2,3\},\{1,2,4\},\{1,2,5\},\{1,3,4\},\{1,3,5\},$$
$$\{1,4,5\},\{2,3,4\},\{2,3,5\},\{2,4,5\},\{3,4,5\}\}.$$

Of these ten ability sets, the Alpern-Chen ordering (b, c, a) has the highest reliability on nine of the ten ability sets. Table 4.2 describes how the entries for the column $D = 6$ in Table 4.1 have been calculated. For each ordering (column), one checks the rows (ability sets) for which that ordering is optimal. The bottom row gives the fraction of the ten ability sets for which the ordering is optimal, which is what we call its optimality fraction. This fraction may sum to more than one due to ties in reliability. Note that in all cases the Alpern-Chen ordering (b, c, a) gives the highest in each column.

From Table 4.1, we can also see that the optimality fraction ϕ decreases monotonically as the deck size increases. This is due to an increasing gap between lower ability and higher ability within a jury given the increasing size of the deck. there are more instances where a juror herds in larger deck sizes, reducing the optimality fraction of the sub-optimal voting orderings. In Table 4.7, it can be seen that the optimality fraction for each voting orderings keeps on decreasing until it eventually fluctuates around a stable value. For instance, the voting ordering (a, b, c) fluctuates around 0.006 and 0.007 for deck size $24 \leq D \leq 52$. Another interesting observation is that the sum of ϕ gets closer to 1 as the deck size increases. However, as the deck size keeps on increasing beyond 52, we conjecture that this value will not converge to 1 because we still consider ability sets $\{a, b, D-1\}$ which cause 4 out of 6 voting ordering (highest ability votes first or second) to always be optimal. As the deck size increases, there are more of such ability sets. Therefore there will always be ties in reliability and the percentage of overlap is proportional to the deck size.

Ability sets \ Orderings	(a,b,c)	(a,c,b)	(b,a,c)	(b,c,a)	(c,a,b)	(c,b,a)
{1,2,3}	0	✓	0	0	0	0
{1,2,4}	0	✓	0	✓	0	0
{1,2,5}	0	✓	0	✓	✓	✓
{1,3,4}	0	0	0	✓	0	0
{1,3,5}	0	✓	0	✓	✓	✓
{1,4,5}	0	✓	0	✓	✓	✓
{2,3,4}	0	0	0	✓	0	0
{2,3,5}	0	✓	0	✓	✓	✓
{2,4,5}	0	✓	0	✓	✓	✓
{3,4,5}	0	✓	0	✓	✓	✓
Counts	0	8	0	9	6	6
Fraction	0	0.8	0	0.9	0.6	0.6

Table 4.2: Optimality fraction for deck size $D = 6$, $(a < b < c)$

Furthermore, we use Figure 4.2 to compare the optimality fractions for all deck sizes, where the vertical axis gives the optimality fraction and the horizontal axis gives the deck size. The legend on the right side of the graph sorts the voting orders from the highest to the lowest. Note that we show D a multiple of 4 on the x-axis for better display of the figure. The reader can refer to Table 4.7 in the Appendix to check that the result is consistent across any $D \leq 52$.

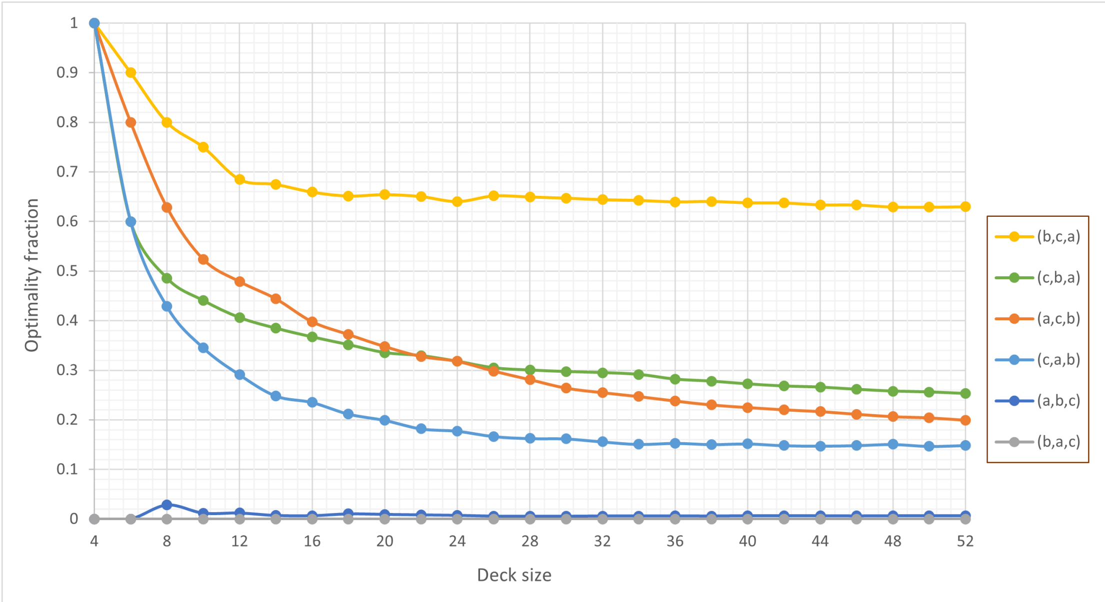

Figure 4.2: Optimality fractions of the six voting orderings

It is shown in Figure 4.2 that the Alpern-Chen voting order is always the highest after deck size 4. When the deck size is near 52, we can see a prevalent ranking on the six voting orderings. The seniority ordering (c, b, a) is the second highest. Meanwhile (b, a, c) line is at exact height of 0, which means that it is never optimal.

Proposition 11. *Given any deck of size $D \leq 52$, the Alpern-Chen ordering (b, c, a) has the highest optimality fraction. Furthermore, ordering (b, a, c) is never optimal.*

4.4.3 On average reliability

For each deck size $D \leq 52$, we calculate the average reliability $\bar{Q}$ for the six voting orders r. The average is taken over all ability sets for D. The results are drawn in Figure 4.3.

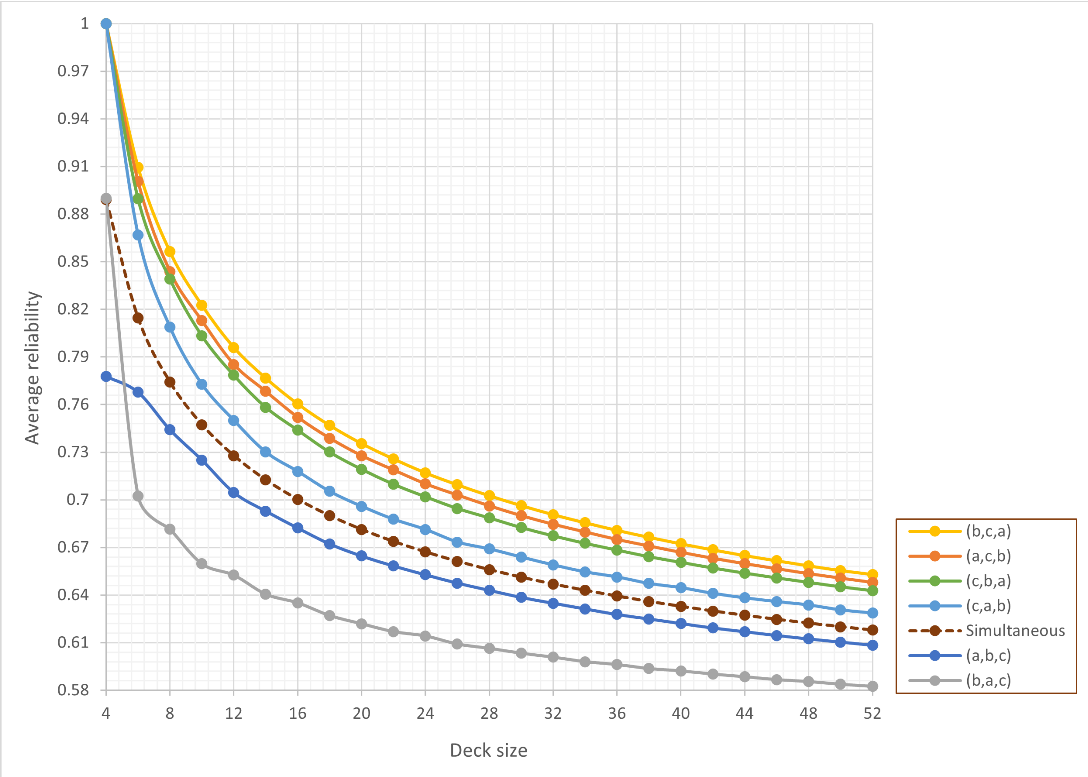

Figure 4.3: Average reliability of the six voting orderings

In Figure 4.3, the legend on the right side shows the six ability orders sorted from the highest average reliability to the lowest. It can be seen that for $8 \leq D \leq 52$ the ordering of the six voting orders stays the same. The Alpern Chen ordering is again always the highest in terms of the average reliability $\bar{Q}$. The second highest is (a, c, b) followed by the seniority ordering (c, b, a). For use in Section 4.5, we have added the average reliability for *simultaneous* voting in the dashed line. We show in Table 4.3 some simple cases to the reader. The curious reader is encouraged to check the full data through Table 4.8 in the Appendix.

Average reliability $\bar{Q}$							
Deck size / Voting order	$D = 4$	6	...	26	...	50	52
(a, b, c)	$7/9 = 0.78$	0.768	...	0.648	...	0.610	0.608
(a, c, b)	1	0.901	...	0.703	...	0.651	0.648
(b, a, c)	$8/9 = 0.89$	0.703	...	0.609	...	0.584	0.582
(b,c,a)	**1**	**0.910**	...	**0.710**	...	**0.655**	**0.653**
(c, a, b)	1	0.867	...	0.673	...	0.631	0.628
(c, b, a)	1	0.890	...	0.695	...	0.645	0.643

Table 4.3: Average reliabilities $\bar{Q}$ for deck size up to 52

Note that when $D = 4$ there is only one ability set $\{1, 2, 3\}$, so the average reliability $\bar{Q}$ is the same as reliability Q for this ability set. The reliabilities for the six orderings were calculated in Example 1. Our proposition follows easily from Table 4.3 and Figure 4.3.

Proposition 12. *Given any deck of size $D \leq 52$, the Alpern-Chen ordering (b, c, a) has an average reliability larger than any other voting order. Furthermore, for $8 \leq D \leq 52$, the average reliability of the six voting orders is the same as indicated by the order of the legends, namely $(b, c, a) > (a, c, b) > (c, b, a) > (c, a, b) > (a, b, c) > (b, a, c)$.*

4.5 Sequential vs. Simultaneous Voting Scheme

Up to now we have been analyzing sequential voting *internally*, comparing the reliability of different sequential voting orders for a fixed set of abilities. In this section, we analyze sequential voting *externally*, by comparing its reliability with the more familiar *simultaneous voting scheme.* This is the scheme studied by [29], where each juror votes based solely on his own private information (here, the cards he samples), without any knowledge of the votes of other jurors. We could equally well call this independent voting. We compare the reliability of sequential voting in the optimal (Alpern-Chen) ordering to that of simultaneous voting. One might think that sequential voting is *always* superior, as jurors have more indirect information about the sealed card. It turns out, as we show in this section, that indeed sequential voting is *generally* superior, but not *always*. The problem with sequential voting, especially for a large jury, is that a high ability juror who is not an oracle (and hence sometimes gets it wrong) may unduly influence subsequent jurors through herding, so that their information is not used at all.

To see this in a simple but large setting, suppose that there is a deck of size D, one juror of ability 3 and all the rest ($n-1$ of them) of ability 1. Suppose the juror of ability 3 goes first in a sequential vote. He has some probability $p = p_3$ that at least two of the cards he draws are the other colour to the sealed card, and that he thus votes correctly. (This is essentially his Condorcet value of p.) Then the next juror (and all subsequent ones), who gets a signal which is correct with probability $p_1 < p_3$, will copy the vote of the first juror and hence the majority verdict, for any jury size n, will be correct with probability $p = p_3$ (that is, if the first juror voted correctly). Now we know from the Condorcet Jury Theorem (under the assumption of Owen, Grofman, and Feld [58] that all jurors have p values of at least p_1), that for sufficiently large jury size n, the reliability of the verdict, which approaches to 1, will exceed that of the sequential jury's reliability p_3. So for very large juries, the Condorcet Jury Theorem ensures that simultaneous voting is superior to sequential voting.

Of course, juries are typically small rather than large. To compare the two voting schemes for a three-member jury, we begin by recalling Example 1, which had a deck of two red and two black cards, and three jurors with abilities 1, 2, and 3 voting sequentially in some order. When the juror of ability 3 (called juror 3) votes first or second, his vote (always correct) will be copied by later voting jurors to produce a correct majority verdict, so reliability is 1. The remaining voting orders $(1, 2, 3)$ and $(2, 1, 3)$ were shown in Example 1 to have reliabilities 7/9 and 8/9, respectively. In particular, sequential voting in the optimal Alpern-Chen ordering $(2, 3, 1)$ has reliability 1. Next consider the reliability of simultaneous (independent) voting. By symmetry, we assume the sealed card is R. What is the probability that the verdict is wrong, B? Since the juror of ability 3 is an oracle who will draw two blacks and a red and give the correct vote of R, the verdict can only be B if that is the vote of the other two. Clearly the juror of ability 1 will vote B if and only if he draws a red card, which has probability 1/3. The juror of ability 2 will vote B if he draws one red and one black card and then randomizes to vote B. This has probability $(2/3)\ \ (1/2) = 1/3$. So the reliability of simultaneous voting in this example is $1 - (1/3)^2 = 8/9$. This is less than the perfect reliability 1 of sequential voting in the optimal (Alpern-Chen) ordering. (It is however better than sequential voting in the suboptimal ability order $(2, 1, 3)$.)

We now give an example where the comparative reliability of the two voting schemes can go the other way: When there are three jurors of common ability 1, simultaneous voting is better. To make the following proof clearer, we assume that the jurors vote in the same order using either scheme, but only in the sequential scheme are they told the votes of the earlier jurors. We use *black* and *red* for the color of the cards drawn, and B and R for the votes and the verdict. For example, we might say 'the second juror draws r but votes R because he copies the first juror'. The simplest ability set for this comparison is when the jurors have the common ability 1, they each draw a single card from the deck. We remove our assumption of distinct abilities here to allow for a very small number of drawn cards. Note that the following result is proved without calculating any reliabilities, just using direct qualitative comparisons.

Theorem 9. *For any deck size $D \geq 4$ and ability set $\{1, 1, 1\}$, simultaneous voting has a higher reliability than sequential voting. Furthermore, for any total private information (draws of card for all players), simultaneous voting is at least as likely as sequential voting to have the correct verdict (N); for some total private information vectors, it is strictly more likely.*

Proof. We prove the second assertion, which implies the first. We divide the private information vector into three cases. In the first two cases, both voting schemes give the same verdict. In the last case, simultaneous voting always gives the verdict most likely to be true (given all the cards drawn), while sequential voting sometimes give the verdict which is less likely to be true.

Case 1: draws red,red,_ (or black,black,_). By symmetry, we can assume first two jurors draw a red. Voting simultaneously, the first two jurors vote $BB\_$ (with verdict $V = B$). The same is true if the second juror sees the vote of the first juror. So in these cases, both schemes give the same verdict.

Case 2: red,black,red (or black,red,black). By symmetry, we assume the cards drawn by the three jurors are re, bl, re. For the simultaneous scheme, voting is BRB and the verdict is $V = B$. With sequential voting, the first juror votes B. Then the second juror (who knows one card of each color has been drawn) randomizes and equiprobably votes R or B. In either case, the third juror votes B. So while the voting sequence may not be the same as for simultaneous voting, the verdict $V = B$ is the same.

Case 3: red,black,black (or black,red,red). Again we assume $red, black, black$ by symmetry. Note that in this case it is more likely that the sealed card is $N = R$. This more likely state of Nature R is the verdict for simultaneous voting, which will be B, R, R. Next consider sequential voting. The first juror votes B and the second juror randomizes between R and B. With positive probability, he votes B, giving the sequence B, B which gives verdict $V = B$. So with positive probability we get the verdict B, which is the verdict less likely to be correct. So with these draws, the verdict of the jury voting sequentially is less likely to be correct than the jury voting simultaneously.

This establishes the second part of the proof, as the three cases cover all possibilities. The reliability of each voting scheme is the sum of the products of their conditional reliability (given the card draws) multiplied by the probability of these draws. Since the probabilities are the same (and positive for all), the comparative results stated above for conditional reliability establishes the claim about reliability in the first sentence.

$\square$

For those readers preferring algebraic forms for the above proof, it can be calculated that the when $p = (D/2) / (D-1)$, the number of cards of opposite color to the sealed card divided by the size of the remaining deck, the reliability of sequential voting is given by $\left(p + 3p^2 - 2p^3\right)/2$ and the reliability of simultaneous voting is $p^3 + 3p^2(1-p)$. The difference is given by

$$\left(p + 3p^2 - 2p^3\right)/2 - \left(p^3 + 3p^2(1-p)\right) = (1/2)\ p\ (2p-1)\ (p-1),$$

which is positive for $1/2 < p < 1$, or equivalently $D \geq 4$.

The reader might now think that for perfectly homogeneous juries (k, k, k), simultaneous voting always has a higher reliability than sequential voting. The following example shows that this is not the case.

Proposition 13. *Suppose we have a deck of $D = 4$ cards (two red, two black) and three jurors of common ability 2. Then sequential voting has a higher reliability (21/27) than simultaneous voting (20/27).*

Proof. First consider sequential voting. If the first two votes are the same, say Y, then that is the majority verdict. Suppose first two votes are distinct, call them X, Y. The first vote X might come from a tie, but if the second voter gets a tie, he copies and votes X. So the vote of Y of juror 2 is certainly correct, as two cards of the same color imply the sealed card is the opposite color (Y). It follows that the third juror will also vote Y. Thus verdict depends on first two votes only, actually it is equal to the vote of the second juror. So the only way sequential voting gives the wrong verdict (say R when the sealed card is $N = B$) is when the first two jurors both draw one card of each color and the first juror randomizes incorrectly to vote R. The probability of drawing one card of each color is $2/3$, so the probability of an incorrect verdict is $(2/3)^2\,(1/2) = 2/9$, so the reliability of sequential voting is $1 - 2/9 = 7/9 = 20/27$.

Next consider simultaneous (independent) voting. The only way a juror can vote incorrectly is if she draws one card of each color and then randomizes the wrong way. This has probability $q = (2/3)\,(1/2) = 1/3$, so $p = 2/3$. Condorcet voting with $p = 2/3$ has reliability $p^3 + 3p^2q = 20/27$, the probability that two or three jurors get the correct signal (opposite color to sealed card).

$\square$

So we have seen that the Condorcet Jury Theorem shows that for large juries, where the reliability of simultaneous juries goes to one, herding may lead to a limiting reliability of sequential voting which is less than one. For juries of size three, we have shown the ability set $\{1, 2, 3\}$ of Example 1 has sequential reliability (in the optimal Alpern-Chen ordering) higher than simultaneous. For perfectly homogeneous juries $\{k, k, k\}$, the inequality of reliabilities can go either way: simultaneous better for $k = 1$ but sequential better for $k = 2$.

In the rest of this section we will show that sequential is generally better and that the ability sets where simultaneous is better tend to be relatively homogeneous. Mainly we will show the results of computer calculations.

4.5.1 Which voting scheme is generally more reliable?

As we have established that for certain deck sizes and ability sets either sequential voting or simultaneous voting can be more reliable. That is, there is no result stating that one of these two schemes is always better. However, in this Subsection, we establish that the Alpern-Chen sequential voting is always better than simultaneous voting in terms of both the average reliability and the optimality fraction.

Average reliability

We recall that the average reliability for a voting scheme, for fixed deck size D, is the average of its reliability over all ability sets with distinct abilities. The average reliability is thus the expected value of the reliability for a randomly chosen jury. Note that in Example 1 where $D = 4$, the only distinct-ability set is $\{1, 2, 3\}$, so the average ability is the same as the reliability for this set. There we showed that sequential voting in the Alpern-Chen order (as well as any order where the juror of ability 3 does not vote last) has reliability 1, and simultaneous voting has reliability 8/9.

For values of D from 6 to 52, the average reliabilities were shown earlier in two of the curves (the highest line for sequential Alpern-Chen and the dashed-line for simultaneous) in Figure 4.3. The calculations illustrated in Figure 4.3 establish the following result.

Proposition 14. *For a three-member jury and deck sizes* D, $4 \leq D \leq 52$, *the average reliability of sequential voting in the Alpern-Chen order is higher than that of simultaneous voting. The difference is always at least* 0.035.

Superiority fraction

We now compare sequential and simultaneous voting, for fixed deck size D, in terms of the fraction of ability sets for which each gives a higher reliability. In particular let the *superiority fraction* $\hat{\phi}(D)$ denote the fraction of ability sets for which simultaneous voting has higher or equal reliability that sequential voting. Recall from our re-analysis above of Example 1 for simultaneous voting in the first part of this section that for $D = 4$ there is only one ability set $\{1, 2, 3\}$ and sequential voting has the higher reliability, so $\hat{\phi}(4) = 0$. By a small amount of computation, we find that $\hat{\phi}(D) = 0$ for all $D < 12$. The computations for $D = 12, 14, \ldots, 52$ are shown in Table 4.9 in the Appendix, leading to the following result. for most ability sets, sequential voting has higher reliability than simultaneous voting, in the following precise sense.

Proposition 15. *The superiority fraction* $\hat{\phi}(D)$ *satisfies the following*

1. *For* D *satisfying* $4 \leq D \leq 10$, $\hat{\phi}(D) = 0$.
2. *For* D *satisfying* $12 \leq D \leq 52$, $\hat{\phi}(D) \leq 0.0268$.

4.5.2 Which ability sets do better with each scheme?

In Section 4.5.1, we asked for *how many* ability sets simultaneous voting has a higher reliability than the Alpern-Chen sequential voting. Now we ask *for which* ability sets simultaneous voting is better. We are not concerned with giving a *list* of these sets, but rather a characterization of them. It turns out that the property of a jury that most determines which voting scheme is more reliable is the heterogeneity of jurors' abilities. We can measure this by the standard deviation of the ability set or the *spread* $\delta = y - x$, the difference between its highest ability y and its lowest ability x. Higher heterogeneity generally means that sequential voting is more likely to be superior. First, we consider only *middle centered* ability sets, those with the middle ability $m = D/2$. These sets are easily illustrated in a two dimensional array, as in Figure 4.4. Then, we relax the restriction on the middle ability, but still consider the spread as the homogeneity measure. Note that in this analysis we revert to our usual restriction that the abilities are distinct, unlike the special common-ability examples of $\{1, 1, 1\}$ and $\{2, 2, 2\}$ discussed at the beginning of the section for ease of presentation.

Spread analysis for centered ability sets

One way to illustrate locations of these ability sets is to produce a two dimensional (x, y) array for *centered ability sets.* More specifically, we condsider ability sets of the form $\{x, m, y\}$, $m = D/2$, with $x < m < y$, where the middle ability is $m = D/2$. Therefore, we can also use integer pair (x, y) to represent the ability set $\{x, m, y\}$. If for this ability set simultaneous voting has a higher reliability than sequential voting in the Alpern-Chen order (m, y, x), we label (x, y) with a small dot. We call such an ability set *dotted* *ability set*. This is drawn in Figure 4.4 for the standard deck size $D = 52$, $m = 26$.

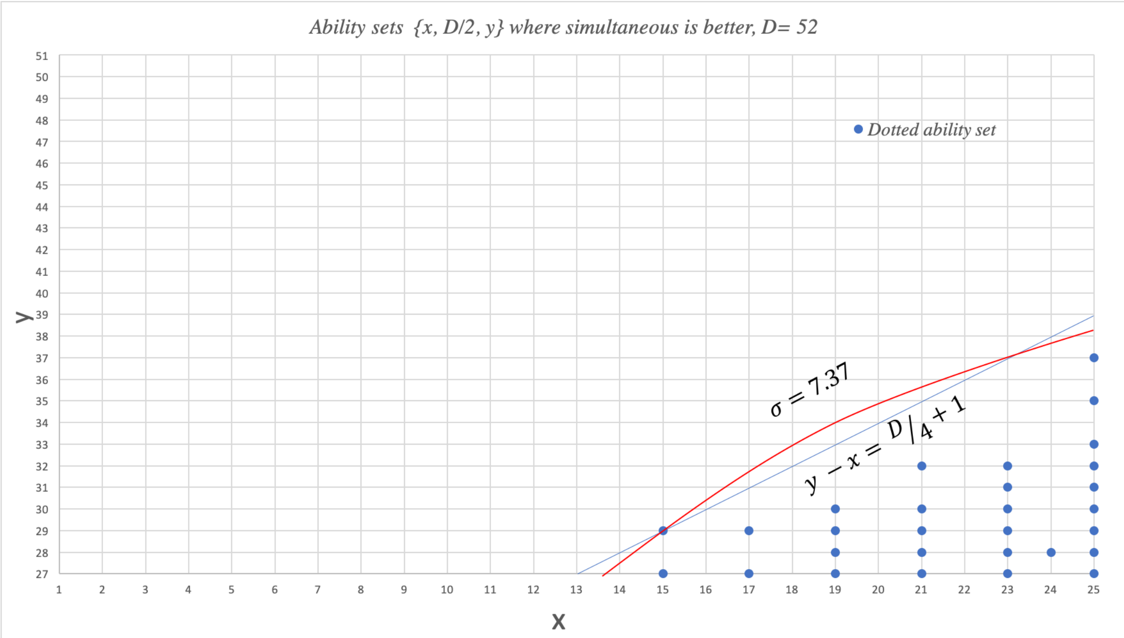

Figure 4.4: Scatter plot of dotted ability sets $\{x, m, y\}$ for $D = 52$.

Note that all the dotted points lie in the triangle under the line $d = y - x = D/4 + 1 = 14$, which contains the 1/8 most homogeneous points (ability sets) with respect to the *spread* measure $y - x$, the difference between the highest and lowest abilities. We also draw a line of constant standard deviation going through the point of ability set $\{15, 26, 29\}$, namely $\sigma = 7.37$ and show that all the dotted points are below this line. Also note that the most homogeneous point $(25, 27)$, corresponding to the ability set $\{25, 26, 27\} = \{m - 1, m, m + 1\}$, is dotted. Note that this point is not dotted for Example 1 with $D = 4$ and unique ability set $\{1, 2, 3\}$, where sequential voting is superior. However it is dotted for sufficiently large deck size D. Also note that when $D = 52$, most of the dotted points have both coordinates odd - this changes to all such points for smaller deck sizes. Recall that jurors with odd abilities cannot draw an equal number of cards of each color – ties are impossible. The lack of ties seems to favor simultaneous voting. The generalizations of observations regarding Figure 4.4

($D = 52$) that hold for other values of D are listed in the following proposition .The data for establishing this result for $D < 52$ is in Figure 4.7 in the Appendix.

Proposition 16. *For $D \equiv 2m \leq 52$, consider ability sets of the form $A = \{x, m, y\}$ with $1 \leq x < m < y < D$. We say that A is* ***dotted*** *(as in Figure 4.4) if the reliability of simultaneous voting with abilities A is greater than or equal to the reliability of sequential voting in the Alpern-Chen order. Then*

1. *If $D \leq 10$ or D is not divisible by 4, then there are no dotted points, that is, the Alpern-Chen sequential voting is always more reliable.*
2. *Suppose D satisfies $12 \leq D \leq 52$ and is divisible by 4. Then all dotted ability sets A are in the triangular set $y \leq x + D/4 + 1$ (with the lowest $1/8$ of spread $y-x$). In particular, the most homogeneous ability set $\{m-1, m, m+1\}$ is dotted. Furthermore if $D \leq 32$ then all dotted points (ability sets) have both x and y odd.*

Spread analysis for general ability sets

We now consider the usual deck size $D = 52$ and consider arbitrary ability sets $\{a, b, c\}$ with $a < b < c$, where the spread is given by $\delta = c - a$. In Figure 4.5, we plot for each value of δ from 2 to 20 the number of ability sets (blue) and the number of these for which the sequential voting is superior to simultaneous voting (total ability sets deducted by the number of dotted ability sets, indicated by orange line). Although δ can go up to $51 - 1 = 50$, it turns out that for $\delta \geq 16$, none of the ability sets are dotted, so we plot only up to 20. For example, when $\delta = 2$ there are 50 ability sets (of the form $\{x, x+1, x+2\}$) and 23 of them are dotted (have higher reliability when voting simultaneously).

These results are consistent with the observation of Ben-Yashar and Nitzar [21], in another model, that in order to produce high reliability with simultaneous voting, a high degree of homogeneity of abilities is required.

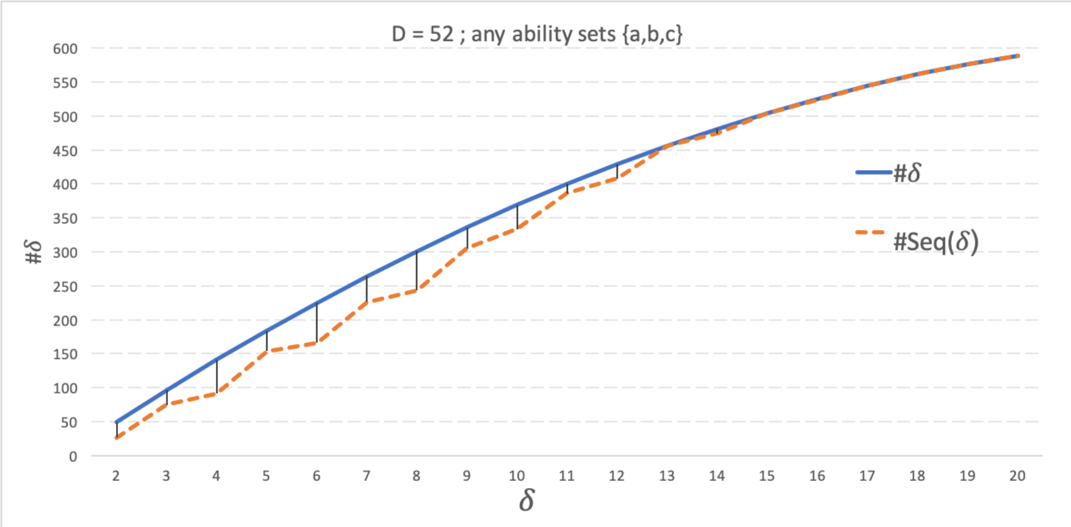

Figure 4.5: Relationship between δ, $\#\delta$, and $\#seq(\delta)$ for fixed $D = 52$.

4.5.3 Reliability and heterogeneity

Kanazawa [44] has shown that the reliability of Condorcet simultaneous voting, for juries of fixed mean of the p value, is increasing in the standard deviation σ of the jury. Here, we show that this is true in terms of the mean and standard deviation of the ability set for both sequential and simultaneous reliability. The additional reliability of sequential over simultaneous voting is also increasing in heterogeneity (as measured by σ).

We illustrate these observations in Figures 4.6 by fixing deck size at $D = 52$ and consider juries with ability sets which have a fixed mean of 26. For each such ability set $A = \{a, b, c\}$ we plot two points: $(\sigma_A, Q(b, c, a))$ and $(\sigma_A, Q_{sim}(A))$, both on the same vertical line with horizontal coordinate σ_A. The horizontal coordinate σ goes from 1 (for the single ability set $\{25, 26, 27\}$) to 25 (for the single ability set $\{1, 26, 51\}$). From Proposition 16, part 2, we know that for $\sigma = 1(\{25, 26, 27\}$ is dotted) the blue point is above the orange point. Note that for any ability set where the highest ability c is 51, the reliability of the sequential voting is 1. There are 13 such ability sets, of the form $(a, b) = (x, 27 - x)$, for $x = 1, \ldots, 13$. These are the 13 points at height (reliability) 1.

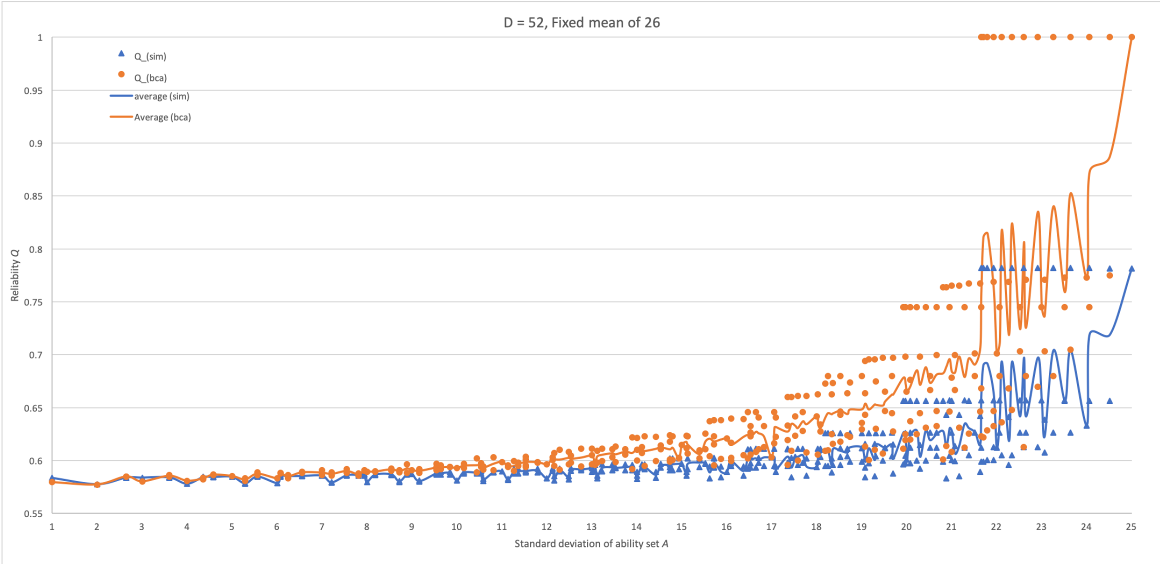

Figure 4.6: Plots of $Q_{sim}(A)$ and $Q(b, c, a)$ against σ of ability sets $\{a, b, c\}$
.

Our observations about reliability increasing with heterogeneity are based on the pattern of the orange dots (sequential reliability) and the blue triangles (simultaneous reliability) in Figure 4.6. We also plot the line corresponding to the averages of the reliabilities for each standard deviation for both the orange dots and the blue triangles. When the standard deviation is larger than 22, we have 13 ability sets containing the juror of ability 51 (reliability is 1 for sequential voting and around 0.78 for simultaneous). This is why the lines are erratic (the higher peak when averaging with reliability 1) between the standard deviation of 22 and 25. These lines increase as the standard deviation becomes larger given that the mean of the ability sets is fixed to 26. This also gives support to the observation of Grofman and Feld [37] that diversity of opinion is a means of arriving at the correct collective decision under democratic majoritarian rules. It can also be seen in the figure that when the standard deviation is very small, the blue line is above the orange line. For instance when the standard deviation is 1, this corresponds to the single ability set $\{25, 26, 27\}$ where the blue dot is higher than the orange dot. As the standard deviation increases, the difference between these two lines also increases with sequential voting (orange line) consistently produces higher reliability than the simultaneous voting. This indicates that the relative advantage of sequential over simultaneous voting increases with heterogeneity. This result goes in the same direction as Theorem 5 of Alpern and Chen [4].

4.6 Average Reliability For $n = 5$

Up to now we have considered the optimal voting ordering for a jury of size three, for which we have definitive results. In this section, we consider sequential jury voting when the jury consists of five jurors. We are still able to use our exhaustive calculation method to obtain the average reliability $\bar{Q}(r)$ for all orderings $r \in \Pi_5$.

Voting orderings	$D = 12$	$D = 14$	$\ldots$	$D = 30$
$r^1 = (d, a, e, c, b)$	0.822	0.801	$\ldots$	0.718
$r^2 = (d, b, e, c, a)$	0.817	0.798	$\ldots$	0.717
$r^3 = (c, a, e, d, b)$	0.815	0.795	$\ldots$	0.715
$r^4 = (c, b, e, d, a)$	0.802	0.785	$\ldots$	0.713
$r^5 = (d, c, e, b, a)$	0.801	0.783	$\ldots$	0.710

Table 4.4: Average reliabilities of the highest five orderings

Of these $\Pi_5 = 120$ orderings, Table 4.4 gives the average reliabilities of the top five orderings, for various deck sizes. For sizes starting with $D = 12$, these remain the same. The full data is provided in Table 4.10 in the Appendix. So we have the following result.

Proposition 17. *Given a deck size between 12 and 30, and a five-member jury of abilities* $\{a, b, c, d, e\}$ *with* $0 < a < b < c < d < e < D$*, the voting orders with the five highest average reliabilities are, in decreasing order,* r^1 *to* r^5 *as labeled in Table 4.4. For comparison with the Alpern-Chen ordering of three jurors, note that for all these orderings, the most able juror (ability* e*) votes in the middle position (third).*

Table 4.5 lists the top five voting orders for average reliability. The same five are top for all deck sizes from 12 to 30. It also shows the average ability v_i of the i^{th}

voter / Ordering	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}
r^1 (Best)	d	a	e	c	b
r^2	d	b	e	c	a
r^3	c	a	e	d	b
r^4	c	b	e	d	a
r^5	d	c	e	b	a
v_i = average	$v_1 = \frac{3d+2c}{5}$	$v_2 = \frac{2a+2b+c}{5}$	$v_3 = e$	$v_4 = \frac{b+2c+2d}{5}$	$v_5 = \frac{3a+2b}{5}$

Table 4.5: Average ability of i^{th} juror to vote, among top five orderings

juror to vote, for the top five voting orders. We now wish to compare our results for juries of size five with our earlier results (and those of Alpern and Chen [2]) for juries of size three. To do this, we partition the five jurors in voting order into three groups: Early, Middle and Late (E, M, L). If we had six jurors, this would be simple, with two in each group. To convert five to three, we must put the second voter partly into E and partly into M . Similar partition applies for the fourth voter. To have the same fractional number of jurors in each group, we put the second voter fractionally 2/3 into E and 1/3 into M, and similarly for the fourth voter. This puts 5/3 voters into each of the three groups, with total weight $3(5/3) = 5$ for the three groups, as required. Since the fractional weight of each group is thus 5/3, we must divide by 5/3 to get the average abilities E, M, L for the three group (for the top five ability orders) as

$$\begin{aligned} E &= \frac{3}{5}\left(v_1 + \frac{2}{3}v_2\right), \\ M &= \frac{3}{5}\left(\frac{1}{3}v_2 + v_3 + \frac{1}{3}v_4\right), and \\ L &= \frac{3}{5}\left(\frac{2}{3}v_4 + v_5\right). \end{aligned}$$

For comparison between the average abilities of the three groups, we calculate

$$25(E - L) = 5(d - a) + 4(c - b) > 0, \text{ and}$$

$$25(M - E) = 15e - (7d + 5c + b + 2a) \geq 15(e - d) > 0.$$

Thus, we have shown the following

Proposition 18. *Given any deck size* $12 \leq D \leq 30$*, for any five-member jury with abilities* $0 < a < b < c < d < e < D$*, and any of the top five orderings in terms of average reliability, we have that*

$$L < E < M,$$

This says that early voters have middle ability, the late voters have lowest ability and the middle voters have the highest ability — precisely the Alpern-Chen ordering for three groups.

4.7 Concluding Remarks

The paper of [2] gives us a simple model of sequential majority jury voting between two equiprobable states of Nature. Their paper shows that when jurors of differing abilities vote sequentially, the reliability of the majority verdict depends on the voting order and the best one is the Alpern-Chen ordering of middle, high, low. In that paper, the notion of ability is defined as a variable in terms of the quality of a juror's private information. In this paper, we introduce a new measurement of a juror's ability, which is simply the number of cards he is allowed to sample from a predetermined deck of cards, in order to guess the one which has been removed from the deck.

We have shown that in our sealed card model, the Alpern-Chen ordering generally has the highest reliability. For decks of size up to 16, this ordering always maximizes the reliability of the majority verdict. For any medium size decks ($D \leq 52$), the Alpern-Chen Ordering is the best one for more juries than any other voting order. We also find that for each deck size, the average reliability of the Alpern-Chen ordering, taken over all possible juries, is higher than for any other ordering.

Having established the optimal ordering, we are able to fix this ordering and make reliability comparisons with other voting schemes. We compare the reliability of the sequential voting scheme with the more familiar simultaneous voting scheme. We showed that in general, sequential voting using the Alpern Chen ordering is better than simultaneous voting in terms of both the average reliability and the optimality fraction. We further asked for which ability sets, if exist, where simultaneous voting is better than Alpern Chen sequential voting. To do this, we define a spread of the set of juror abilities with smaller spread representing more homogeneous juries. We have found that for sufficiently homogeneous juries, simultaneous voting can sometimes have higher reliability than sequential voting. On the other hand, when juries are sufficiently heterogeneous, sequential voting is more reliable. In a similar vein, we have found that the reliability of a jury of fixed average ability is increasing in its heterogeneity.

For a jury of size five we show that the five highest voting orders, in terms of average reliability, all have the most able juror voting in the middle (third) position. This is an analog of the Alpern-Chen ordering. We also show that these orderings, when grouped into early, middle and late voters have the following: early voters have middle average ability, middle voters have highest average ability and late voters have lowest average ability.

4.8 Appendix

		Voting Ordering										
D	Ability sets	$(b,c,a)^*$	$>$	(b,a,c)	$\mid$	(a,c,b)	$>$	(a,b,c)	$<$	(c,b,a)	$>$	(c,a,b)
8	$\{2,4,6\}$	**0.7878**		0.6834		0.7551		0.6933		0.7469		0.7143
	$\{2,4,6\}$	**0.6882**		0.6352		0.6705		0.6403		0.6718		0.6429
	$\{2,4,8\}$	**0.7751**		0.6526		0.7531		0.6628		0.7275		0.7222
10	$\{2,6,8\}$	**0.8016**		0.6429		0.7531		0.6987		0.7538		0.7222
	$\{4,6,8\}$	**0.8016**		0.7109		0.7751		0.7197		0.7606		0.7285
	$\{2,4,6\}$	**0.6383**		0.6075		0.6296		0.6107		0.6328		0.6082
	$\{2,4,8\}$	**0.6860**		0.6172		0.6723		0.6228		0.6575		0.6515
	$\{2,4,10\}$	**0.7686**		0.6344		0.7521		0.6453		0.7294		0.7273
	$\{2,6,8\}$	**0.7007**		0.6082		0.6741		0.6469		0.6801		0.6515
	$\{2,6,10\}$	**0.7863**		0.6082		0.7521		0.6714		0.7361		0.7273
12	$\{2,8,10\}$	**0.8099**		0.6515		0.7521		0.7022		0.7578		0.7273
	$\{4,6,8\}$	**0.7015**		0.6592		0.6883		0.6637		0.6867		0.6586
	$\{4,6,10\}$	**0.7863**		0.6794		0.7686		0.6869		0.7363		0.7308
	$\{4,8,10\}$	**0.8099**		0.6620		0.7686		0.7177		0.7628		0.7318
	$\{6,8,10\}$	**0.8099**		0.7257		0.7863		0.7342		0.7691		0.7379
	$\{2,4,6\}$	**0.6125**		0.5893		0.6032		0.5916		0.6085		0.5874
	$\{2,4,8\}$	**0.6422**		0.5959		0.6330		0.5997		0.6224		0.6166
	$\{2,4,10\}$	**0.6851**		0.6053		0.6743		0.6115		0.6608		0.6573
	$\{2,4,12\}$	**0.7647**		0.6223		0.7515		0.6338		0.7308		0.7308
	$\{2,6,8\}$	**0.6522**		0.5874		0.6350		0.6177		0.6413		0.6166
	$\{2,6,10\}$	**0.6959**		0.5874		0.6743		0.6305		0.6674		0.6573

*The boldface under (b,c,a) corresponds to Proposition 8, the one " $<$ " covers Proposition 9, and the three " $>$ "s indicate Proposition 10.

	$\{2,6,12\}$	**0.7778**	0.5874	0.7515	0.6549	0.7342	0.7308
	$\{2,8,10\}$	**0.7087**	0.6166	0.6767	0.6512	0.6853	0.6573
	$\{2,8,12\}$	**0.7935**	0.6166	0.7515	0.6768	0.7421	0.7308
	$\{2,10,12\}$	**0.8155**	0.6573	0.7515	0.7047	0.7605	0.7308
14	$\{4,6,8\}$	**0.6514**	0.6286	0.6433	0.6315	0.6472	0.6234
	$\{4,6,10\}$	**0.6956**	0.6398	0.6856	0.6439	0.6677	0.6621
	$\{4,6,12\}$	**0.7778**	0.6602	0.7647	0.6673	0.7351	0.7308
	$\{4,8,10\}$	**0.7089**	0.6257	0.6879	0.6643	0.6905	0.6630
	$\{4,8,12\}$	**0.7935**	0.6284	0.7647	0.6889	0.7422	0.7308
	$\{4,10,12\}$	**0.8155**	0.6667	0.7647	0.7169	0.7641	0.7308
	$\{6,8,10\}$	**0.7103**	0.6732	0.6991	0.6775	0.6965	0.6695
	$\{6,8,12\}$	**0.7935**	0.6943	0.7778	0.7009	0.7424	0.7371
	$\{6,10,12\}$	**0.8155**	0.6750	0.7778	0.7290	0.7688	0.7388
	$\{8,10,12\}$	**0.8155**	0.7352	0.7935	0.7435	0.7748	0.7444
	$\{2,4,6\}$	**0.5949**	0.5764	0.5875	0.5782	0.5919	0.5734
	$\{2,4,8\}$	**0.6157**	0.5814	0.6089	0.5842	0.6007	0.5952
	$\{2,4,10\}$	**0.6435**	0.5877	0.6353	0.5920	0.6262	0.6224
	$\{2,4,12\}$	**0.6847**	0.5968	0.6759	0.6037	0.6630	0.6615
	$\{2,4,14\}$	**0.7621**	0.6137	0.7511	0.6256	0.7333	0.7333
	$\{2,6,8\}$	**0.6214**	0.5734	0.6075	0.5985	0.6168	0.5952
	$\{2,6,10\}$	**0.6511**	0.5734	0.6366	0.6069	0.6323	0.6224
	$\{2,6,12\}$	**0.6932**	0.5734	0.6760	0.6197	0.6673	0.6615
	(2,6,14)	**0.7725**	0.5734	0.7511	0.6439	0.7338	0.7333
	$\{2,8,10\}$	**0.6597**	0.5952	0.6388	0.6223	0.6468	0.6224
	$\{2,8,12\}$	**0.7025**	0.5952	0.6759	0.6357	0.6745	0.6615
	$\{2,8,14\}$	**0.7841**	0.5952	0.7511	0.6612	0.7377	0.7333
	$\{2,10,12\}$	**0.7143**	0.6224	0.6786	0.6542	0.6888	0.6615

	$\{2,10,14\}$	**0.7986**	0.6224	0.7511	0.6807	0.7464	0.7333
	$\{2,12,14\}$	**0.8195**	0.6615	0.7511	0.7065	0.7623	0.7333
	$\{4,6,8\}$	**0.6249**	0.6081	0.6189	0.6101	0.6219	0.6017
	$\{4,6,10\}$	**0.6518**	0.6157	0.6453	0.6184	0.6326	0.6273
	$\{4,6,12\}$	**0.6932**	0.6267	0.6848	0.6308	0.6681	0.6643
16	$\{4,6,14\}$	**0.7725**	0.6472	0.7621	0.6542	0.7338	0.7333
	$\{4,8,10\}$	**0.6564**	0.6034	0.6434	0.6335	0.6517	0.6281
	$\{4,8,12\}$	**0.7025**	0.6049	0.6853	0.6464	0.6747	0.6651
	$\{4,8,14\}$	**0.7841**	0.6078	0.7621	0.6711	0.7385	0.7333
	$\{4,10,12\}$	**0.7139**	0.6305	0.6879	0.6650	0.6930	0.6660
	$\{4,10,14\}$	**0.7986**	0.6332	0.7621	0.6905	0.7464	0.7333
	$\{4,12,14\}$	**0.8195**	0.6698	0.7621	0.7165	0.7651	0.7333
	$\{6,8,10\}$	**0.6607**	0.6415	0.6543	0.6442	0.6569	0.6343
	$\{6,8,12\}$	**0.7024**	0.6532	0.6942	0.6567	0.6751	0.6701
	$\{6,8,14\}$	**0.7841**	0.6748	0.7725	0.6805	0.7394	0.7339
	$\{6,10,12\}$	**0.7148**	0.6381	0.6967	0.6753	0.6977	0.6715
	$\{6,10,14\}$	**0.7986**	0.6425	0.7725	0.6998	0.7466	0.7340
	$\{6,12,14\}$	**0.8195**	0.6776	0.7725	0.7261	0.7687	0.7341
	$\{8,10,12\}$	**0.7166**	0.6825	0.7065	0.6867	0.7034	0.6773
	$\{8,10,14\}$	**0.7986**	0.7041	0.7841	0.7102	0.7469	0.7419
	$\{8,12,14\}$	**0.8195**	0.6843	0.7841	0.7367	0.7731	0.7439
	$\{10,12,14\}$	**0.8195**	0.7418	0.7986	0.7500	0.7789	0.7492

Table 4.6: Reliability Q for $D \leq 16$ and even ability, Proposition 8, 9, and 10

$D \backslash r$	(a,b,c)	(a,c,b)	(b,a,c)	(b,c,a)	(c,a,b)	(c,b,a)	*Sum*
4	0.000	1.000	0.000	1.000	1.000	1.000	4.000
6	0.000	0.800	0.000	0.900	0.600	0.600	2.900
8	0.029	0.629	0.000	0.800	0.429	0.486	2.371
10	0.012	0.524	0.000	0.750	0.345	0.440	2.071
12	0.012	0.479	0.000	0.685	0.291	0.406	1.873
14	0.007	0.444	0.000	0.675	0.248	0.385	1.759
16	0.007	0.398	0.000	0.659	0.235	0.367	1.666
18	0.010	0.372	0.000	0.651	0.212	0.351	1.597
20	0.009	0.348	0.000	0.654	0.199	0.335	1.546
22	0.008	0.328	0.000	0.650	0.182	0.329	1.498
24	0.007	0.318	0.000	0.640	0.177	0.318	1.461
26	0.006	0.298	0.000	0.652	0.167	0.305	1.427
28	0.005	0.281	0.000	0.650	0.162	0.300	1.399
30	0.005	0.264	0.000	0.647	0.162	0.297	1.375
32	0.006	0.255	0.000	0.644	0.156	0.295	1.356
34	0.006	0.247	0.000	0.643	0.151	0.291	1.338
36	0.006	0.238	0.000	0.640	0.153	0.282	1.319
38	0.006	0.230	0.000	0.641	0.150	0.278	1.305
40	0.007	0.225	0.000	0.638	0.152	0.272	1.293
42	0.007	0.220	0.000	0.637	0.148	0.268	1.280
44	0.006	0.217	0.000	0.634	0.147	0.266	1.270
46	0.006	0.211	0.000	0.633	0.148	0.261	1.260
48	0.006	0.206	0.000	0.629	0.151	0.258	1.250
50	0.007	0.204	0.000	0.629	0.147	0.256	1.242
52	0.007	0.200	0.000	0.630	0.148	0.253	1.238

Table 4.7: Optimality Fraction ϕ for $D \leq 52$

D \ r	(a,b,c)	(a,c,b)	(b,a,c)	(b,c,a)	(c,a,b)	(c,b,a)	*simultaneous*
4	0.7778	1	0.8889	1	1	1	0.888999
6	0.7678	0.9006	0.7025	0.9096	0.86695	0.8896	0.8146
8	0.744128	0.843855	0.681486	0.85628	0.8088	0.839024	0.774127
10	0.725061	0.812857	0.659651	0.822672	0.772841	0.803265	0.747348
12	0.704583	0.785297	0.652501	0.796018	0.750056	0.77844	0.727762
14	0.692853	0.768276	0.640532	0.776651	0.730099	0.758285	0.712546
16	0.682228	0.751981	0.635073	0.760459	0.717897	0.743916	0.700239
18	0.672185	0.738815	0.627091	0.746923	0.705303	0.730055	0.689993
20	0.664594	0.727777	0.621887	0.735475	0.695876	0.719259	0.681275
22	0.658326	0.718764	0.616938	0.725789	0.687751	0.709833	0.67373
24	0.652772	0.710041	0.614211	0.716997	0.681314	0.70195	0.667109
26	0.647514	0.702963	0.60912	0.709481	0.673368	0.69456	0.661234
28	0.642993	0.696167	0.60641	0.702621	0.669046	0.688619	0.655971
30	0.638499	0.689947	0.603337	0.696407	0.663914	0.682538	0.651218
32	0.634848	0.684518	0.600855	0.690692	0.658975	0.677337	0.646897
34	0.631287	0.679688	0.598015	0.685594	0.654575	0.672615	0.642943
36	0.62775	0.674934	0.596187	0.680797	0.651382	0.668366	0.639308
38	0.624959	0.670851	0.59369	0.676422	0.647283	0.664191	0.635949
40	0.622059	0.666824	0.592177	0.672309	0.644644	0.660627	0.632833
42	0.619261	0.663077	0.590193	0.66847	0.641029	0.657066	0.62993
44	0.616911	0.659695	0.588524	0.664908	0.638197	0.653857	0.627218
46	0.614529	0.656505	0.586695	0.661634	0.63587	0.650837	0.624676
48	0.612361	0.653408	0.58549	0.65846	0.633715	0.648017	0.622286
50	0.610391	0.650569	0.583772	0.655492	0.630596	0.645235	0.620035
52	0.608398	0.647872	0.582477	0.652719	0.62865	0.64278	0.617907

Table 4.8: Average reliability $\bar{Q}$ for $D \leq 52$

D	12	14	16	18	20	22	24	26
$\hat{\phi}(D)$	0.0242	0.021	0.022	0.0235	0.0268	0.0248	0.0226	0.0204
D	28	30	32	34	36	38	40	42
$\hat{\phi}(D)$	0.0191	0.0175	0.0165	0.0163	0.0163	0.0156	0.0165	0.0173
D	44	46	48	50	52			
$\hat{\phi}(D)$	0.0168	0.0178	0.0179	0.0181	0.0187			

Table 4.9: The superiority fraction of ability sets in which $Q_{sim} > Q(b, c, a)$ given D

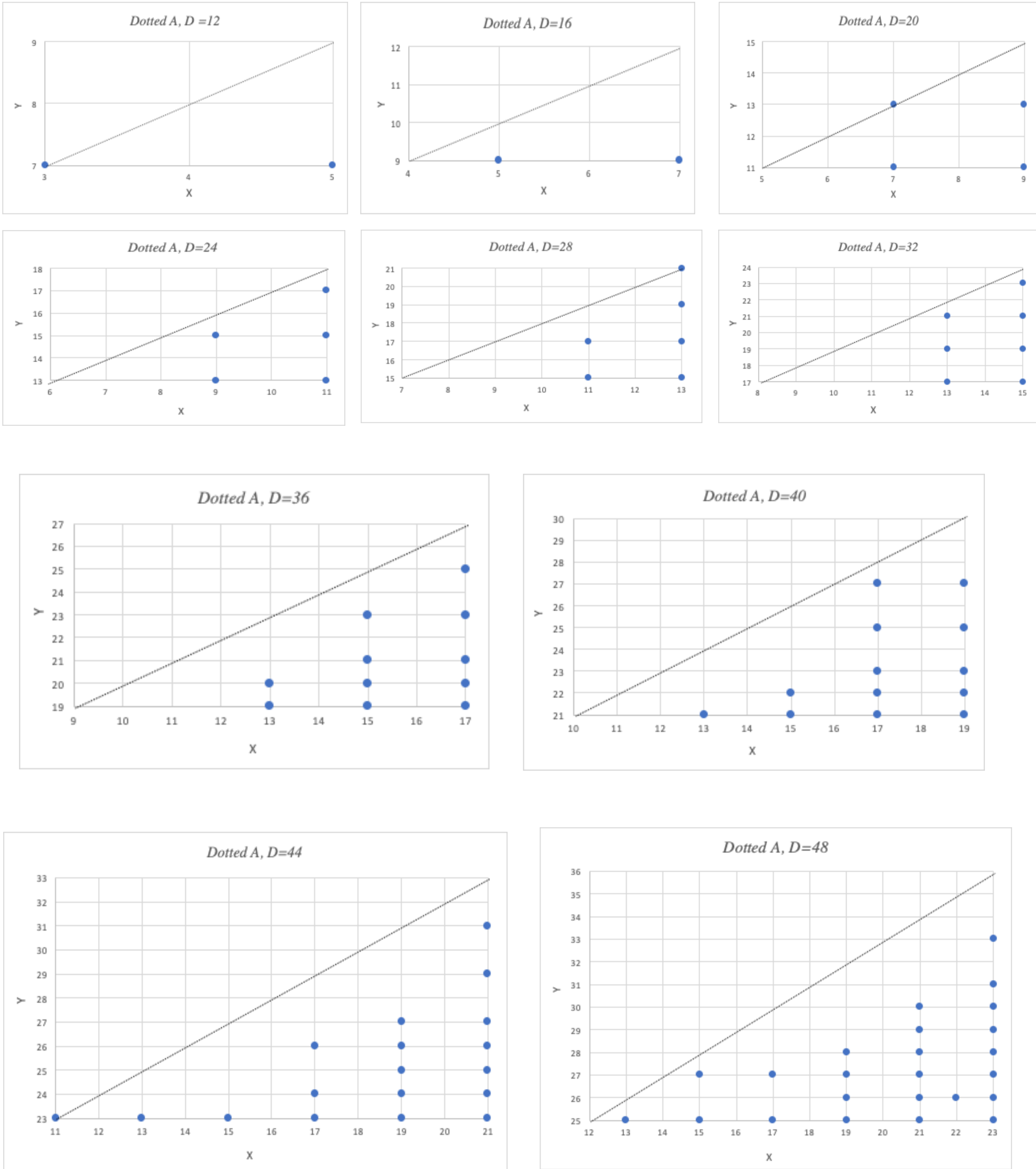

Figure 4.7: Graphs of Dotted A for $12 \leq D < 52$.

$r \backslash D$	12	14	16	18	20	22	24	26	28	30
$(d, a, e, c, b)^{\dagger}$	0.822	0.802	0.785	0.771	0.759	0.747	0.737	0.733	0.724	0.718
(d, b, e, c, a)	0.817	0.797	0.781	0.768	0.757	0.746	0.736	0.732	0.723	0.717
(c, a, e, d, b)	0.814	0.795	0.779	0.766	0.755	0.744	0.734	0.729	0.721	0.715
(c, b, e, d, a)	0.802	0.785	0.771	0.760	0.750	0.741	0.732	0.727	0.719	0.714
(d, c, e, b, a)	0.801	0.783	0.769	0.756	0.745	0.736	0.727	0.722	0.714	0.709
(a, b, c, d, e)	0.724	0.711	0.699	0.689	0.681	0.674	0.665	0.662	0.655	0.651
(a, b, c, e, d)	0.770	0.754	0.740	0.729	0.720	0.712	0.703	0.699	0.692	0.687
(a, b, d, c, e)	0.715	0.700	0.688	0.678	0.670	0.663	0.655	0.652	0.646	0.643
(a, b, d, e, c)	0.782	0.768	0.756	0.745	0.736	0.728	0.718	0.714	0.706	0.702
(a, b, e, c, d)	0.742	0.726	0.713	0.701	0.692	0.685	0.676	0.672	0.666	0.662
(a, b, e, d, c)	0.784	0.770	0.758	0.747	0.738	0.730	0.722	0.717	0.710	0.705
(a, c, b, d, e)	0.732	0.718	0.705	0.695	0.686	0.679	0.671	0.667	0.660	0.656
(a, c, b, e, d)	0.762	0.745	0.732	0.721	0.712	0.704	0.696	0.692	0.686	0.681
(a, c, d, b, e)	0.723	0.708	0.696	0.685	0.677	0.670	0.662	0.659	0.653	0.648
(a, c, d, e, b)	0.794	0.778	0.764	0.752	0.743	0.734	0.725	0.720	0.712	0.707
(a, c, e, b, d)	0.719	0.707	0.697	0.689	0.682	0.675	0.669	0.666	0.661	0.658
(a, c, e, d, b)	0.785	0.771	0.759	0.748	0.739	0.731	0.723	0.718	0.711	0.706
(a, d, b, c, e)	0.737	0.722	0.711	0.701	0.693	0.685	0.677	0.673	0.667	0.663
(a, d, b, e, c)	0.771	0.756	0.743	0.732	0.723	0.715	0.706	0.702	0.695	0.690
(a, d, c, b, e)	0.731	0.718	0.706	0.696	0.688	0.682	0.673	0.669	0.663	0.659
(a, d, c, e, b)	0.789	0.771	0.757	0.745	0.735	0.726	0.717	0.712	0.705	0.700
(a, d, e, b, c)	0.742	0.728	0.717	0.708	0.701	0.694	0.688	0.684	0.677	0.674
(a, d, e, c, b)	0.770	0.756	0.746	0.736	0.728	0.721	0.714	0.710	0.703	0.699

†These are the highest five orderings as stated in Proposition 17

(a,e,b,c,d)	0.769	0.755	0.742	0.732	0.722	0.715	0.707	0.703	0.695	0.691
(a,e,b,d,c)	0.761	0.747	0.735	0.725	0.716	0.709	0.701	0.697	0.690	0.686
(a,e,c,b,d)	0.767	0.753	0.741	0.731	0.721	0.714	0.706	0.702	0.695	0.691
(a,e,c,d,b)	0.762	0.749	0.739	0.729	0.720	0.712	0.704	0.700	0.693	0.689
(a,e,d,b,c)	0.773	0.759	0.747	0.737	0.728	0.720	0.712	0.708	0.701	0.696
(a,e,d,c,b)	0.765	0.752	0.741	0.731	0.722	0.715	0.707	0.703	0.696	0.691
(b,a,c,d,e)	0.692	0.685	0.676	0.668	0.660	0.657	0.648	0.645	0.637	0.634
(b,a,c,e,d)	0.754	0.744	0.732	0.721	0.711	0.704	0.695	0.691	0.683	0.679
(b,a,d,c,e)	0.680	0.670	0.661	0.653	0.647	0.645	0.636	0.634	0.627	0.625
(b,a,d,e,c)	0.775	0.760	0.748	0.738	0.728	0.721	0.712	0.707	0.699	0.695
(b,a,e,c,d)	0.729	0.711	0.697	0.686	0.676	0.671	0.661	0.657	0.650	0.646
(b,a,e,d,c)	0.796	0.779	0.765	0.752	0.742	0.733	0.723	0.719	0.711	0.705
(b,c,a,d,e)	0.716	0.704	0.694	0.684	0.677	0.670	0.663	0.659	0.653	0.649
(b,c,a,e,d)	0.736	0.722	0.711	0.702	0.694	0.689	0.680	0.677	0.669	0.665
(b,c,d,a,e)	0.717	0.705	0.693	0.683	0.675	0.668	0.661	0.657	0.652	0.648
(b,c,d,e,a)	0.795	0.779	0.765	0.754	0.744	0.735	0.726	0.721	0.714	0.709
(b,c,e,a,d)	0.704	0.696	0.688	0.681	0.676	0.670	0.666	0.664	0.658	0.656
(b,c,e,d,a)	0.789	0.775	0.762	0.751	0.742	0.734	0.725	0.720	0.713	0.708
(b,d,a,c,e)	0.742	0.727	0.714	0.704	0.695	0.688	0.679	0.675	0.668	0.664
(b,d,a,e,c)	0.756	0.742	0.730	0.720	0.712	0.705	0.697	0.693	0.685	0.681
(b,d,c,a,e)	0.734	0.720	0.707	0.696	0.688	0.681	0.672	0.668	0.661	0.657
(b,d,c,e,a)	0.786	0.770	0.756	0.744	0.735	0.726	0.717	0.713	0.706	0.701
(b,d,e,a,c)	0.720	0.712	0.704	0.697	0.692	0.686	0.681	0.678	0.671	0.669
(b,d,e,c,a)	0.772	0.758	0.747	0.737	0.729	0.722	0.714	0.711	0.703	0.699
(b,e,a,c,d)	0.765	0.754	0.741	0.731	0.721	0.714	0.705	0.702	0.695	0.690
(b,e,a,d,c)	0.760	0.749	0.736	0.726	0.717	0.709	0.701	0.697	0.690	0.686
(b,e,c,a,d)	0.763	0.749	0.738	0.728	0.719	0.713	0.705	0.701	0.694	0.690

(b,e,c,d,a)	0.763	0.750	0.739	0.730	0.721	0.714	0.705	0.702	0.694	0.690
(b,e,d,a,c)	0.765	0.754	0.744	0.735	0.726	0.719	0.712	0.708	0.701	0.697
(b,e,d,c,a)	0.758	0.745	0.736	0.727	0.718	0.712	0.704	0.700	0.693	0.689
(c,a,b,d,e)	0.712	0.698	0.686	0.677	0.669	0.660	0.654	0.650	0.643	0.640
(c,a,b,e,d)	0.748	0.733	0.720	0.710	0.702	0.693	0.685	0.681	0.675	0.671
(c,a,d,b,e)	0.692	0.682	0.672	0.664	0.657	0.651	0.645	0.642	0.637	0.633
(c,a,d,e,b)	0.789	0.774	0.759	0.748	0.738	0.729	0.719	0.714	0.706	0.701
(c,a,e,b,d)	0.704	0.690	0.678	0.670	0.663	0.656	0.650	0.647	0.642	0.638
(c,b,a,d,e)	0.700	0.689	0.679	0.671	0.663	0.656	0.650	0.646	0.641	0.638
(c,b,a,e,d)	0.723	0.710	0.699	0.692	0.684	0.678	0.671	0.667	0.662	0.658
(c,b,d,a,e)	0.700	0.690	0.681	0.672	0.665	0.659	0.652	0.649	0.644	0.640
(c,b,d,e,a)	0.786	0.772	0.760	0.750	0.740	0.732	0.723	0.718	0.710	0.706
(c,b,e,a,d)	0.699	0.689	0.680	0.672	0.666	0.659	0.655	0.652	0.648	0.645
(c,d,a,b,e)	0.745	0.730	0.716	0.705	0.695	0.688	0.679	0.676	0.669	0.664
(c,d,a,e,b)	0.765	0.750	0.737	0.726	0.716	0.708	0.700	0.696	0.689	0.684
(c,d,b,a,e)	0.738	0.723	0.710	0.699	0.689	0.682	0.673	0.671	0.664	0.659
(c,d,b,e,a)	0.773	0.758	0.745	0.734	0.725	0.716	0.708	0.704	0.697	0.692
(c,d,e,a,b)	0.738	0.727	0.717	0.708	0.700	0.694	0.688	0.685	0.679	0.675
(c,d,e,b,a)	0.771	0.757	0.745	0.735	0.727	0.721	0.713	0.710	0.703	0.699
(c,e,a,b,d)	0.758	0.747	0.736	0.727	0.719	0.711	0.704	0.700	0.694	0.689
(c,e,a,d,b)	0.753	0.742	0.732	0.722	0.715	0.708	0.701	0.696	0.690	0.686
(c,e,b,a,d)	0.756	0.744	0.734	0.725	0.717	0.710	0.703	0.698	0.692	0.688
(c,e,b,d,a)	0.756	0.745	0.735	0.726	0.718	0.712	0.704	0.699	0.693	0.688
(c,e,d,a,b)	0.753	0.745	0.737	0.730	0.722	0.716	0.710	0.705	0.700	0.696
(c,e,d,b,a)	0.742	0.735	0.727	0.721	0.713	0.707	0.701	0.697	0.691	0.687
(d,a,b,c,e)	0.713	0.702	0.691	0.683	0.675	0.667	0.661	0.657	0.651	0.648
(d,a,b,e,c)	0.749	0.737	0.724	0.715	0.707	0.699	0.692	0.687	0.681	0.677

(d,a,c,b,e)	0.707	0.697	0.688	0.680	0.673	0.666	0.660	0.656	0.650	0.647
(d,a,c,e,b)	0.771	0.756	0.742	0.732	0.723	0.714	0.705	0.701	0.694	0.689
(d,a,e,b,c)	0.753	0.737	0.722	0.710	0.700	0.692	0.683	0.678	0.671	0.667
(d,b,a,c,e)	0.711	0.700	0.689	0.680	0.673	0.666	0.659	0.656	0.649	0.646
(d,b,a,e,c)	0.734	0.725	0.712	0.703	0.696	0.689	0.682	0.678	0.671	0.668
(d,b,c,a,e)	0.715	0.703	0.693	0.684	0.676	0.669	0.663	0.659	0.653	0.649
(d,b,c,e,a)	0.779	0.763	0.749	0.739	0.730	0.722	0.712	0.708	0.701	0.696
(d,b,e,a,c)	0.724	0.715	0.701	0.692	0.684	0.677	0.671	0.668	0.662	0.659
(d,c,a,b,e)	0.722	0.710	0.700	0.690	0.681	0.674	0.667	0.663	0.657	0.653
(d,c,a,e,b)	0.749	0.733	0.721	0.712	0.704	0.698	0.689	0.686	0.679	0.675
(d,c,b,a,e)	0.723	0.710	0.699	0.689	0.680	0.672	0.666	0.662	0.656	0.652
(d,c,b,e,a)	0.765	0.748	0.736	0.726	0.718	0.710	0.702	0.698	0.692	0.687
(d,c,e,a,b)	0.744	0.730	0.719	0.710	0.701	0.694	0.687	0.683	0.677	0.673
(d,c,e,b,a)	0.801	0.783	0.769	0.756	0.745	0.736	0.727	0.722	0.714	0.709
(d,e,a,b,c)	0.738	0.729	0.722	0.715	0.706	0.700	0.695	0.692	0.687	0.684
(d,e,a,c,b)	0.726	0.720	0.714	0.708	0.698	0.693	0.689	0.686	0.681	0.679
(d,e,b,a,c)	0.751	0.741	0.729	0.722	0.711	0.706	0.701	0.698	0.691	0.688
(d,e,b,c,a)	0.743	0.735	0.724	0.717	0.706	0.701	0.695	0.693	0.686	0.683
(d,e,c,a,b)	0.749	0.741	0.732	0.725	0.715	0.711	0.704	0.702	0.696	0.692
(d,e,c,b,a)	0.740	0.734	0.723	0.717	0.707	0.703	0.696	0.694	0.688	0.684
(e,a,b,c,d)	0.746	0.732	0.720	0.711	0.704	0.697	0.690	0.687	0.681	0.677
(e,a,b,d,c)	0.732	0.720	0.711	0.703	0.696	0.690	0.683	0.680	0.674	0.671
(e,a,c,b,d)	0.743	0.727	0.716	0.707	0.699	0.693	0.686	0.683	0.677	0.674
(e,a,c,d,b)	0.740	0.726	0.715	0.707	0.700	0.694	0.687	0.683	0.678	0.675
(e,a,d,b,c)	0.746	0.730	0.719	0.710	0.702	0.696	0.688	0.685	0.678	0.675
(e,a,d,c,b)	0.749	0.732	0.720	0.711	0.703	0.697	0.689	0.686	0.679	0.676
(e,b,a,c,d)	0.745	0.732	0.721	0.713	0.704	0.698	0.692	0.688	0.680	0.678

(e,b,a,d,c)	0.740	0.726	0.717	0.709	0.701	0.694	0.688	0.682	0.676	0.675
(e,b,c,a,d)	0.731	0.719	0.711	0.705	0.698	0.692	0.687	0.682	0.676	0.674
(e,b,c,d,a)	0.755	0.740	0.730	0.722	0.713	0.705	0.699	0.693	0.687	0.684
(e,b,d,a,c)	0.744	0.728	0.719	0.711	0.703	0.697	0.691	0.686	0.679	0.677
(e,b,d,c,a)	0.758	0.741	0.730	0.720	0.711	0.703	0.696	0.692	0.685	0.682
(e,c,a,b,d)	0.755	0.742	0.731	0.723	0.714	0.707	0.700	0.697	0.691	0.687
(e,c,a,d,b)	0.769	0.754	0.743	0.735	0.725	0.715	0.707	0.703	0.696	0.693
(e,c,b,a,d)	0.737	0.727	0.718	0.712	0.704	0.697	0.692	0.689	0.683	0.680
(e,c,b,d,a)	0.765	0.751	0.742	0.734	0.724	0.715	0.707	0.703	0.696	0.693
(e,c,d,a,b)	0.765	0.750	0.738	0.729	0.719	0.711	0.704	0.699	0.692	0.688
(e,c,d,b,a)	0.764	0.748	0.737	0.728	0.718	0.709	0.701	0.696	0.689	0.685
(e,d,a,b,c)	0.750	0.744	0.736	0.730	0.723	0.716	0.711	0.707	0.701	0.698
(e,d,a,c,b)	0.761	0.752	0.743	0.736	0.728	0.718	0.713	0.708	0.702	0.698
(e,d,b,a,c)	0.740	0.737	0.729	0.723	0.716	0.710	0.705	0.701	0.695	0.692
(e,d,b,c,a)	0.764	0.754	0.745	0.737	0.728	0.719	0.713	0.708	0.701	0.697
(e,d,c,a,b)	0.748	0.740	0.732	0.725	0.718	0.712	0.705	0.702	0.696	0.692
(e,d,c,b,a)	0.752	0.743	0.733	0.726	0.718	0.710	0.703	0.699	0.693	0.688

Table 4.10: The average reliability $\hat{Q}$ between $12 \leq D \leq 30$ for a jury of size 5

Chapter 5

Summary and Future Works

In conclusion, this thesis is a collection of three essays on search and pursuit games and jury voting problems. In both these areas I incorporated the analysis of sequential learning. In Chapter 2 Section 2.8, we answer a natural question whether it is possible that after repeated search and escape, the capture probability in the locations could in theory be learnt by the searcher and the hider. Our contribution using stochastic games provide a simplistic understanding that having to learn the capture probability is favourable to the prey. Moreover, when these probabilities are high, variability (heterogeneity) of the locations in terms of their capture probabilities favour the predator. This is the first analysis into understanding the effect of learning in search games. The same discussion using the deterministic game form in Chapter 3 Section 3.4 gives support to the findings established by the stochastic version. In Chapter 4, our study investigates the optimal voting ordering of a heterogeneous jury in an open sequential vote for binary alternatives. Different from previous work in this area, the ordering of votes between jurors with different ability plays an important role in determining the reliability of the verdict. The reliability is the probability that the majority vote agrees with the true state of Nature. Given the signal and ability of the first juror, the subsequent jurors receive information about the votes. This information includes the abilities of the previous voters, and the *a priori* probability of the binary alternatives. Through Bayes law, subsequent jurors update the *a priori* probability and votes honestly according to his signals. This is a learning process through information aggregation between the jurors. Our results in this study show that the Alpern Chen ordering (median ability first, highest ability second, and lowest ability last) produces the highest average reliability and optimality fraction in comparison with the other voting orderings.

The collection of results in the three papers presented in this thesis, although mostly theoretical, provides clear contributions to the central objectives in Operations Research. The research objective of the three papers is to determine the optimal policy for practical problems given the circumstances. In understanding the behaviour of animals during hunting season, Theorem 6 under Chapter 2 for instance provides implications for the question of when to keep hiding or to leave the lair and by what route. Our modified model in Chapter 3 improves upon the work of [36] where the time to search each location can now be a non-constant parameter. Our study on the optimal voting orderings for heterogeneous jury in a sequential majority voting is within the area of optimization. Much of our findings presented in this work (Chapter 4) serve as heuristic yet robust suggestions for the leadership team within an organization to best convenes a group of experts into reaching the final decision through a majority vote.

Finally, several potential avenues for extension can be envisaged to our proposed models and solutions. The study of search and pursuit games as well as optimal ordering under sequential voting are both relatively new within their respective area of study. Thus, there are huge potential for interdisciplinary joint research to add upon our works. Listed below are the potential future studies for the three essays in this thesis.

- One may argue that in a real-life scenario, the game played between a predator and a prey may not necessarily have to be a zero-sum game as analyzed in Chapter 2. The predator is chasing after it's "dinner" and the prey is running for it's life. We have noted in Section 2.9 that non-zero sum stochastic games have been developed only a few years after zero-sum games were introduced. The games have been proven to be difficult to solve and it's complexity had been increasing rapidly since. Hence, for search and pursuit games, implementing non-zero sum games represent a new and much needed field. In addition, one of the referees reviewing our work for the Journal of Royal Society Interface commented on the feasibility of using Evolutionary Game Theory in further advancing the work in this area. Indeed, our current games are not suitable for Evolutionary Game analysis or Evolutionary Stable Strategy (ESS). Search Games do not have pure strategy saddle points. Thus, the emphasis of our model is that animals can learn these games socially rather than evolutionarily, as shown in [52]. We certainly believe that the evolutionary aspect might be of interest to game theorist and biologist for future translational joint research.

- The three sub-problems we introduced in Chapter 3 can benefit from different approach to solving the deterministic game. One example can be the use of *fictituous play* as demonstrated in [48] through their Best Response Oracle algorithm. In a more general sense, fictitious play is useful to find the game's value in iterative methods, which are easier to compute than analytical methods when the problem gets more complicated. Implementing this algorithm into learning enables us to find the solution to any two players zero-sum games, including the normal form games of search and pursuit.

- We also made strict assumptions for our learning model which could potentially be extended in future studies. First, it is interesting to see if the distribution of the conditional capture probability could be made arbitrary instead of equiprobable. This is difficult to envision in the normal form model. However, in the dynamic version of the learning game, when there are only two locations, this simply means that $g(0)$ will have an initial value of θ, and since the effective capture probability $\hat{p}(0)$ is dependent on $g(0)$, this will no longer be $(a+b)/2$ (randomize between the two locations at time 0) and instead, the searcher will initially search according to the effective capture probability of $\hat{p}(0) = \theta a + (1-\theta)b$. Another possible extension is to make the search time for each location to be exogenous with respect to the hider's strategy. For instance the hider could add layers of obstacles in a location with lower capture probability to make it harder for the searcher to traverse and thus, artificially increasing the search time. This location will then be more attractive for the searcher to allocate his effort. The hider can even hide in other locations, leaving her "artificial" hiding location as a *decoy* to trick the searcher to waste his search time. This analysis of "smart" hider using a decoy in a hiding location has been analysed in a different model between a scatter hoarder and a pilferer [6].

- Our propositions on optimal voting ordering in Chapter 4 can benefit from experimental studies. Our results are based on computations from a finite number of cases using Wolfram's *Mathematica* and MatLab. It would be interesting to test our model with the behaviour research lab in which the jury ordering will be randomized, and the ability of each juror will be determined prior using computer interface test. The purpose of this study is to fill the gap in understanding group's voting behaviour through artificial decision-making scenario, which will be of interest to social scientist or behavioural researcher.

Bibliography

[1] Alpern, S., & Baston, V. J. (2017). The secretary problem with a selection committee : Do conformist committees hire better secretaries. *Management Science*, 63(4), 1184 -1197.

[2] Alpern, S., & Chen, B. (2017a). The importance of voting order for jury decisions by sequential majority voting. *European Journal of Operational Research*, 258(3), 1072–1081.

[3] Alpern, S., & Chen, B. (2017b). Who should cast the casting vote? Using sequential voting to amalgamate information. *Theory and Decision*, 83(2), 259–282.

[4] Alpern, S., & Chen, B. (2020). Optimizing Voting Order on Sequential Juries: A Median Voter Theorem. *arXiv preprint arXiv:2006.14045*.

[5] Alpern, S., Chen, B, Lee,V., & Pan, C.X. (2020). Optimizing Voting Order on Sequential Juries: A Sealed Card Model .Available at SSRN: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3761745.

[6] Alpern, S., Fokkink, R., Lidbetter, T., & Clayton, N. (2012). A search game model of the scatter hoarder's problem. *Journal of the Royal Society Interface*, 9(70), 869–879.

[7] Alpern, S., Fokkink, R., Timmer, M., & Casas, J. (2011). Ambush frequency should increase over time during optimal predator search for prey. *Journal of The Royal Society Interface*, 8(64), 1665–1672.

[8] Alpern, S., Fokkink, R., Gal, S., & Timmer, M. (2013). On search games that include ambush. *SIAM Journal on Control and Optimization*, 51(6), 4544–4556.

[9] Alpern S., & Gal S. (2003). The Theory of Search Games and Rendezvous. *Kluwer Academic Publishers.*Dordrecht, The Netherlands.

[10] Alpern, S., & Gal, S. (2006) *The Theory of Search Games and Rendezvous. Kluwer Academic Publishers.* Dordrecht, The Netherlands.

[11] Gal S., Alpern, S., & Casas, J. (2015). Prey should hide more randomly when a predator attacks more persistently. *Journal of the Royal Society Interface*, 12 (113), 20150861.

[12] Alpern, S., Gal, S., Lee,V., & Casas,J.(2019) A stochastic game model of searching predators and hiding prey. *Journal of the Royal Society Interface*, 16(153), 20190087

[13] Alpern, S., Gal, S., & Solan, E. (2010). A sequential selection game with vetoes. *Games and Economic Behavior*, 68(1), 1–14.

[14] Alpern S., & Lee V. (2020) A Normal Form Game Model of Search and Pursuit. In Ramsey, D.M., & Renault, J. (eds), *Advances in Dynamic Games.* Annals of the International Society of Dynamic Games, 17, 57–73. Birkhäuser, Cham.

[15] Anderson E. J., & Aramendia, M. A. (1990). The search game on a network with immobile hider. *Networks*, 20(7), 817–844.

[16] Arcullus, R. (2013). A discrete search-ambush game with a silent predator. In *Search Theory. A Game Theoretic Perspective*, 249–266. *Springer.* New York, USA.

[17] Avis, D., Rosenberg, G., Savani, R., & Stengel, B. von (2010) Enumeration of Nash Equilibria for Two-Player Games. *Economic Theory*, 42(1), 9-37. Online solver available at http://banach.lse.ac.uk.

[18] Baston V., & Kikuta K (2015). Search games on a network with travelling and search costs. *Int. J. Game Theory*, 44(2), 347–365

[19] Bateman, P. W., & Fleming, P. A. (2014). Switching to Plan B: changes in the escape tactics of two grasshopper species *Acrididae: Orthoptera* in response to repeated predatory approaches.*Behavioral Ecology and Sociobiology*, 68(3), 457–465.

[20] Ben-Gal I., & Kagan E. (2013). Information Search After Static or Moving Targets: Theory and Modern Applications. *Wiley* .New York, USA.

[21] Ben-Yashar, R., & Nitzan, S. (2017). Is diversity in capabilities desirable when adding decision makers? *Theory and Decision*, 82(3), 395—402.

[22] Boland, P.J. (1989). Majority systems and the Condorcet Jury Theorem. *The Statistician*, 38(3), 181–189.

[23] Brams, S.J., & Fishburn, P.C. (2002). Voting procedures. In Arrow, K., Sen, A., & Suzumura, K.(eds), *Handbook of Social Choice and Welfare*, 1, 173-–236.

[24] Brighton, C. H., Thomas, A. L., & Taylor, G. K. (2017). Terminal attack trajectories of peregrine falcons are described by the proportional navigation guidance law of missiles. *Proceedings of the National Academy of Sciences*, 114(51), 13495–13500.

[25] Broom, M. (2013). Interactions Between Searching Predators and Hidden Prey. In *Search Theory. A Game Theoretic Perspective*, 233–248. Springer New York.

[26] Broom, M., & Rychtár, J. (2013). Game-theoretical Models in Biology. *CRC Press.*

[27] Bryce, C. M., Wilmers, C. C., & Williams, T. M. (2017). Energetics and evasion dynamics of large predators and prey: pumas vs. hounds. *PeerJ*, 5, e3701.

[28] Chung,T., Hollinger,G., & Isler,V.(2011) Search and pursuit-evasion in mobile robotics. *Autonomous Robots*, 31(4), 299–316

[29] Condorcet, M.J.A.N.C. (1785). Essai sur l'Application de l'Analyse à la Prababilité des Decisions Rendues à la Pluralité des Voix (Impr. Royale, Paris); reprinted (1972) (Chelsea, New York).

[30] Cooper, W.E., Cooper Jr, W. E., & Blumstein, D. T. (Eds.). (2015). Escaping from Predators: an Integrative View of Escape Decisions. *Cambridge University Press.*

[31] Dekel, E., & Piccione, M. (2000). Sequential voting procedures in symmetric binary elections. *Journal of Political Economy*, 108(1), 34–55.

[32] Djemai, I., Casas, J., & Magal, C. (2004). Parasitoid foraging decisions mediated by artificial vibrations. *Animal Behaviour*, 67(3), 567–571.

[33] Domenici, P., Blagburn, J. M., & Bacon, J. P. (2011). Animal escapology I: theoretical issues and emerging trends in escape trajectories. T*he Journal of Experimental Biology*, 214(15), 2463–2473.

[34] Free, B., McHenry, M. J., & Paley, D. (2019). Probabilistic analytical modelling of predator–prey interactions in fishes. *J. R. Soc. Interface*, 16 (150), 20180873.

[35] Friedman, D., & Sinervo, B. (2016). Evolutionary Games in Natural, Social, and Virtual Worlds. *Oxford University Press.*

[36] Gal, S., & Casas, J. (2014). Succession of hide–seek and pursuit–evasion at heterogeneous locations. *Journal of the Royal Society Interface*, 11(94), 20140062.

[37] Grofman, B., & Feld, S.L. (1988). Rousseau's General Will: A Condorcetian Perspective. *American Political Science Review*, 82(2), 567-–576

[38] Hall, C. F. (1865). Arctic Researches, and Life Among the Esquimaux: Being the Narrative of an Expedition in Search of Sir John Franklin, in the Years 1860, 1861, and 1862. *Harper & Brothers.*

[39] Hall, M.G., & Brace, P. (1994). The vicissitudes of death by decree: Forces influencing capital punishment decision making in state supreme courts. *Social Science Quarterly* 75(1).

[40] Hoeffding, W. (1956). On the distribution of the number of successes in independent trials. *The Annals of Mathematical Statistics*, 27(3), 713–721.

[41] Hohzaki, R. (2016). Search games: Literature and survey. *Journal of the Operations Research Society of Japan*, 59(1), 1-34.

[42] Isaacs, R.(1965).Differential Games. *Wiley.* New York, USA.

[43] Johnson, T.R., Spriggs, J.F., & Wahlbeck, P.J. (2005). Passing and strategic voting on the US Supreme Court. *Law & Society Review*, 39(2), 349–378.

[44] Kanazawa, S. (1998). A brief note on a further refinement of the Condorcet Jury Theorem for heterogeneous groups. *Mathematical Social Sciences*, 35(1), 69–73.

[45] Kellerer, H., Pferschy, U., & Pisinger, D. (2004). Multidimensional knapsack problems. In *Knapsack problems*, 235–283. *Springer.* Berlin, Heidelberg.

[46] Kelly, B. P., Badajos, O. H., Kunnasranta, M., Moran, J. R., Martinez-Bakker, M., Wartzok, D., & Boveng, P. (2010). Seasonal home ranges and fidelity to breeding sites among ringed seals. *Polar Biology*, 33(8), 1095–1109.

[47] Lidbetter, T (2013). Search games with multiple hidder objects. *SIAM J. Control Optim.*, 51(4), 3056–3074.

[48] Hellerstein,L., Lidbetter,T., & Pirutinsky,D.(2019) Solving Zero-sum Games using Best Response Oracles with Applications to Search Games. *Operations Research*, 67(3), 731-743

[49] Lin, K.Y., & Singham, D.I. (2016). Finding a hider by an unknown deadline. *Operations Research Letters* ,44(1), 25–32.

[50] Mangel, M., & Clark, C. W. (1988). Dynamic Modeling in Behavioral Ecology. *Princeton University Press.*

[51] Martin, J., & Lopez, P. (2003). Changes in the escape responses of the lizard *Acanthodactylus erythrurus* under persistent predatory attacks. *Copeia*, 2003(2), 408–413.

[52] Martin, C. F., Bhui, R., Bossaerts, P., Matsuzawa, T., & Camerer, C. (2014) Chimpanzee choice rates in competitive games match equilibrium game theory predictions. *Scientific Reports*, 4, 5182.

[53] Mech, L. D., Smith, D. W., & MacNulty, D. R. (2015). Wolves on the hunt: the behavior of wolves hunting wild prey. *University of Chicago Press.*

[54] Meyhöfer, R., Casas, J., & Dorn, S. (1997). Vibration–mediated interactions in a host–parasitoid system. *Proceedings of the Royal Society of London B: Biological Sciences*, 264(1379), 261–266.

[55] Morice S., Pincebourde S., Darboux F., Kaiser W., & Casas J. (2013). Predator–prey pursuit–evasion games in structurally complex environments. *Integr. Comp. Biol.* 53(5), 767–779.

[56] Morse, P.M., Kimball, G.E. ,& Gass, S.I.(2003). Methods of operations research. *Courier Corporation.*

[57] Ottaviani, M., & Sørensen, P. (2001). Information aggregation in debate: who should speak first? *Journal of Public Economics*, 81(3), 393–421.

[58] Owen, G., Grofman, B., & Feld, S.L. (1989). Proving a distribution-free generalization of the Condorcet jury theorem. *Mathematical Social Sciences*, 17(1), 1–16.

[59] Pincebourde, S., & Casas, J. (2015). Warming tolerance across insect ontogeny: influence of joint shifts in microclimates and thermal limits. *Ecology*, 96(4), 986–997.

[60] Pitchford, J. (2013). Applications of search in biology: some open problems. In *Search Theory: A Game Theoretic Perspective*, 295-303. *Springer*, New York, USA.

[61] Puterman, M. L. (2005). Markov Decision Processes: Discrete Stochastic Dynamic Programming. *Wiley Series in Probability and Statistics.*

[62] Ross, S. M. (1983). Introduction to Stochastic Dynamic Programming. *Academic press.*

[63] Rudebeck, G. (1951). The choice of prey and modes of hunting of predatory birds with special reference to their selective effect. *Oikos*, 3(2), 200–231.

[64] Selten, R. (1988). A note on evolutionarily stable strategies in asymmetric animal conflicts. In *Models of Strategic Rationality*, 67–75. *Springer*. Dordrecht, The Netherlands.

[65] Shapley L.S. (1953) Stochastic games. *Proc Natl Acad Sci USA*, 39(10), 1095–1100.

[66] Sillar, K. T., Picton, L. D., & Heitler, W. J. (2016). The Neuroethology of Predation and Escape. *John Wiley & Sons.*

[67] Solan, E., & Vieille, N. (2015). Stochastic Games: A Perspective. *Proceedings of the National Academy of Sciences of the USA.* 112(10), 13743–13746.

[68] Stephens, D. W., Brown, J. S., & Ydenberg, R. C. (Eds.). (2007). Foraging: Behavior and Ecology. *University of Chicago Press.*

[69] Stephens, D. W., & Krebs, J. R. (1986). Foraging Theory. *Princeton University Press.*

[70] Stirling, I. (1974). Midsummer observations on the behavior of wild polar bears (*Ursus maritimus*). *Canadian Journal of Zoology*, 52(9), 1191-1198.

[71] Weibull, J. (2017). Evolutionary game theory. Lecture Notes. *Delhi Winter School*, December 2017. (http://econdse.org/wp-content/uploads/2014/07/lecture-dws1a.pdf)

[72] Wilson A., Lowe J., Roskilly K., Hudson P., Golabek K., & McNutt J. (2013). Locomotion dynamics of hunting in wild cheetahs. *Nature*, 498(7453), 185–189.

[73] Zoroa, N., Fernández-Sáez, M. J., & Zoroa, P. (2011) A foraging problem: sit-and-wait versus active predation. *Eur. J. Oper. Res.*, 208(2), 131–141.

[74] Zoroa, N., Fernández-Sáez, M. J., & Zoroa, P. (2015). Ambush and active search in multistage predator–prey Interactions. *Journal of Optimization Theory and Applications*, 166(2), 626–643.