BULFER, Andrew Frederick, 1938-
A STUDY OF A DISCRETE PREDICTION AND EVASION PROBLEM.
The Ohio State University, Ph.D., 1970
Engineering, electrical

University Microfilms, A XEROX Company, Ann Arbor, Michigan

71-7408

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED
A STUDY OF A DISCRETE PREDICTION
AND EVASION PROBLEM

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the Graduate
School of The Ohio State University

By

Andrew Frederick Bulfer, B.S., M.Sc.

The Ohio State University
1970

Approved by

K. Hemami
Advisor
Department of Electrical Engineering
ACKNOWLEDGMENTS

The author wishes to express his appreciation to Professor Hooshang Hemami for his help, suggestions, and constructive criticism throughout the time this work was in progress. He is also indebted to Professor Lee J. White whose guidance and advice have proven invaluable, and to his former colleagues at the Johns Hopkins University Applied Physics Laboratory with whom he had many valuable discussions from which the germ of the idea for this dissertation originated. The cooperation of Mrs. Mary Kimball who typed the manuscript is also fully appreciated.

Finally, the author acknowledges with gratitude the essential contribution of his wife, Peggy, whose forbearance, tolerance and continued encouragement helped make this work possible.
November 23, 1938 ..................... Born - Chicago, Illinois

1962 ................................. B.S.E.E., Massachusetts Institute of Technology, Cambridge, Massachusetts

1962-1967 ........................... Systems Analyst, Johns Hopkins University Applied Physics Laboratory, Silver Spring, Maryland

1968-1970 ........................... National Science Foundation Graduate Fellow

1969 ................................. M.Sc., The Ohio State University, Columbus, Ohio

1969-1970 ........................... Teaching Associate, Department of Electrical Engineering, The Ohio State University, Columbus, Ohio

PUBLICATIONS

"Tracking and Future Prediction of an Evading, Goal-Seeking Vehicle."


FIELDS OF STUDY

Major Field: Electrical Engineering

Studied in Modern Control Theory. Professor H. Hemami

Studied in Switching Theory and Logical Design. Professor R. B. McGhee

Studied in Computer and Information Science. Professor L. J. White
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>ii</td>
</tr>
<tr>
<td>VITA</td>
<td>iii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>viii</td>
</tr>
<tr>
<td>Chapter</td>
<td></td>
</tr>
<tr>
<td>I.  INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Description of the Problem</td>
<td></td>
</tr>
<tr>
<td>1.2 Aiming and Evasion and Game Theory</td>
<td></td>
</tr>
<tr>
<td>1.3 Synopsis of the Dissertation</td>
<td></td>
</tr>
<tr>
<td>II. PRIOR WORK</td>
<td>10</td>
</tr>
<tr>
<td>III. PROBLEM FORMULATION</td>
<td>14</td>
</tr>
<tr>
<td>3.1 General Formulation</td>
<td></td>
</tr>
<tr>
<td>3.2 The Case of Linear Equations and Origin-Independent Payoff</td>
<td></td>
</tr>
<tr>
<td>3.3 The Austere Problem</td>
<td></td>
</tr>
<tr>
<td>3.4 Isaacs' Game</td>
<td></td>
</tr>
<tr>
<td>3.5 Conclusions</td>
<td></td>
</tr>
<tr>
<td>IV. EXISTENCE OF SOLUTIONS</td>
<td>38</td>
</tr>
<tr>
<td>4.1 Conversion to Normalized Form</td>
<td></td>
</tr>
<tr>
<td>4.2 Proof of Existence</td>
<td></td>
</tr>
<tr>
<td>4.3 Conclusions</td>
<td></td>
</tr>
<tr>
<td>V. THE SUB-NORMALIZED FORM</td>
<td>43</td>
</tr>
<tr>
<td>5.1 The Semi-Normalized Form</td>
<td></td>
</tr>
<tr>
<td>5.2 Example</td>
<td></td>
</tr>
<tr>
<td>5.3 The Sub-Normalized Form</td>
<td></td>
</tr>
<tr>
<td>5.4 Example</td>
<td></td>
</tr>
<tr>
<td>5.5 Advantages of the Sub-Normalized Form</td>
<td></td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>VI.</td>
<td>TECHNIQUES FOR SOLUTION</td>
</tr>
<tr>
<td>6.1</td>
<td>Mixed Strategies</td>
</tr>
<tr>
<td>6.2</td>
<td>Solution by Linear Programming</td>
</tr>
<tr>
<td>6.3</td>
<td>Example</td>
</tr>
<tr>
<td>6.4</td>
<td>Solution by Dynamic Programming</td>
</tr>
<tr>
<td>6.5</td>
<td>Example</td>
</tr>
<tr>
<td>6.6</td>
<td>Summary and Conclusions</td>
</tr>
<tr>
<td>VII.</td>
<td>GRAPH THEORETIC CONSIDERATIONS</td>
</tr>
<tr>
<td>7.1</td>
<td>Formulation as a Graph</td>
</tr>
<tr>
<td>7.2</td>
<td>Symmetry</td>
</tr>
<tr>
<td>7.3</td>
<td>The Path Function</td>
</tr>
<tr>
<td>7.4</td>
<td>Summary and Conclusions</td>
</tr>
<tr>
<td>VIII.</td>
<td>BOUNDS ON THE VALUE</td>
</tr>
<tr>
<td>8.1</td>
<td>Rationale of Bounds on the Value</td>
</tr>
<tr>
<td>8.2</td>
<td>A Technique for Obtaining Bounds</td>
</tr>
<tr>
<td>8.3</td>
<td>Bounds on the Value</td>
</tr>
<tr>
<td>8.4</td>
<td>Summary</td>
</tr>
<tr>
<td>IX.</td>
<td>FINITE MEMORY GAMES</td>
</tr>
<tr>
<td>9.1</td>
<td>Finite Memory Games</td>
</tr>
<tr>
<td>9.2</td>
<td>Solution of the MO-Game</td>
</tr>
<tr>
<td>9.3</td>
<td>Example</td>
</tr>
<tr>
<td>9.4</td>
<td>Solution of the EO-Game</td>
</tr>
<tr>
<td>9.5</td>
<td>Example</td>
</tr>
<tr>
<td>9.6</td>
<td>The El-Game</td>
</tr>
<tr>
<td>9.7</td>
<td>Conclusions</td>
</tr>
<tr>
<td>X.</td>
<td>THE EXACT ANALYTICAL SOLUTION FOR K = ( \lambda = 2 )</td>
</tr>
<tr>
<td>10.1</td>
<td>Optimal Strategy for the Marksman</td>
</tr>
<tr>
<td>10.2</td>
<td>Optimal Strategy for the Evader</td>
</tr>
<tr>
<td>10.3</td>
<td>Proof of Optimality</td>
</tr>
<tr>
<td>10.4</td>
<td>Closed Form Solutions</td>
</tr>
<tr>
<td>10.5</td>
<td>Asymptotic Form in Games of Long Duration</td>
</tr>
<tr>
<td>10.6</td>
<td>Example</td>
</tr>
<tr>
<td>10.7</td>
<td>Summary</td>
</tr>
<tr>
<td>XI.</td>
<td>SUMMARY AND CONCLUSIONS</td>
</tr>
</tbody>
</table>
# APPENDIX

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>172</td>
</tr>
<tr>
<td>B</td>
<td>176</td>
</tr>
<tr>
<td>C</td>
<td>179</td>
</tr>
<tr>
<td>D</td>
<td>186</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>190</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table                                Page
8-1  Upper and lower bounds on the Value of the game.  108
9-1  $W_{E1}$ for various values of $K$ and $\lambda$.  137
10-1 Example of marksman's optimal aiming strategy. 162
10-2 Payoff resulting from the various choices of the evader.  163
10-3 Example of evader's optimal mixed strategy. 163
10-4 Hit probabilities as a function of marksman's aiming points. 164
**LIST OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-1</td>
<td>Example of semi-normalized payoff matrix $P'$.</td>
<td>49</td>
</tr>
<tr>
<td>5-2</td>
<td>Example of sub-normalized payoff matrix $P''$.</td>
<td>55</td>
</tr>
<tr>
<td>6-1</td>
<td>Example of Primal Linear Programming Problem.</td>
<td>70</td>
</tr>
<tr>
<td>6-2</td>
<td>Example of Dual Linear Programming Problem.</td>
<td>71</td>
</tr>
<tr>
<td>6-3</td>
<td>Optimal solution of linear programming example.</td>
<td>72</td>
</tr>
<tr>
<td>7-1</td>
<td>Examples of graphs.</td>
<td>85</td>
</tr>
<tr>
<td>8-1</td>
<td>Bounds on the Value when $\lambda = 2$.</td>
<td>106</td>
</tr>
<tr>
<td>8-2</td>
<td>Bounds on the Value when $K = 2$.</td>
<td>107</td>
</tr>
<tr>
<td>9-1</td>
<td>Example of evader's optimal mixed strategy in MO-Game.</td>
<td>123</td>
</tr>
<tr>
<td>9-2</td>
<td>Tighter upper bound on the Value when $\lambda = 2$.</td>
<td>138</td>
</tr>
<tr>
<td>10-1</td>
<td>State table of the sequential machine $M$.</td>
<td>141</td>
</tr>
<tr>
<td>10-2</td>
<td>State diagram of the sequential machine $M$.</td>
<td>142</td>
</tr>
<tr>
<td>10-3</td>
<td>State transition diagram of Markov Chain $Y$.</td>
<td>147</td>
</tr>
<tr>
<td>10-4</td>
<td>Graphs used in calculating the hit probability $q_{lij}$.</td>
<td>150</td>
</tr>
</tbody>
</table>
CHAPTER I
INTRODUCTION

The general problem of predicting the future course of a mobile hostile enemy who is deliberately maneuvering so as to confound prediction arises in many guises. In the classic military context, it appears in the form of a marksman attempting to aim at a maneuvering target. In the strategic domain, the vital problem is to evaluate quantitatively the future threat posed by a particular enemy so as to effectively counter it. In the world of business, the ability to estimate a competitor's future marketing strategy is often vital to financial success. Even in science and engineering, if one wishes to steer a conservative course he adopts the Murphian viewpoint that nature may be counted upon to do her worst and is therefore an enemy whose actions must be anticipated.

Whatever be the details, the factors each of these situations have in common are the time lag between the current information and that which must be estimated, and the fact that the participants are antagonists each of whose performance can only be improved at the expense of the others'.

This dissertation represents an attempt to isolate and analyze one simple member of this class of prediction problems. Since only a
single version would be studied, an initial decision was made as to which one. The criteria used in making the choice were:

1. It had to embody those two factors above which are the sine qua non of the genus.

2. It should contain as few complicating or confusing details as possible. The idea is to focus on the fundamental structure of this type of problem. Thus a simple problem was desired, one which is devoid of extraneous considerations which could confuse rather than enlighten.

3. It should be well-behaved mathematically. Answers like "no solution exists" are, to the author at least, not very interesting. Such singular behavior can often be avoided by careful formulation.

4. It should somehow be related to past work so that answers may be compared.

The problem which was finally selected is a generalized version of what previous authors have christened: "the problem of aiming and evasion". The problem is one of prediction and evasion. However, since the terminology of "aiming and evasion" is richer and its occurrence within ordinary experience more common, it shall be used rather than that of "prediction and evasion" as a format for the following description of the problem.

1.1 Description of the Problem

The general problem of aiming and evasion, although technically very difficult, is deceptively easy to state in words. There
are two adversaries: a marksman and an evader. The marksman, who is stationary, must aim his weapon, a gun say, so as to hit the evader. The difficulty is that there is a rather long time delay between the moment the gun is fired and the time the bullet reaches its destination. The marksman must therefore predict the evader's future position based on known present and past data and direct his fusillade in such a way that the evader has a high probability of being hit. The evader, on the other hand, is unarmed save for his mobility and his strong survival instinct. His only hope is to dodge and weave continuously (subject of course to dynamic limitations) in such a way that he and a bullet do not arrive at the same spot simultaneously. The marksman's weapon is assumed to be an automatic one capable of continuously firing an inexhaustible supply of bullets. These are assumed to be invisible to the evader so that he must choose his course without knowledge of the marksman's strategy. The problem is to find where the marksman should aim his weapon and how the evader should dodge and weave in order for each to achieve his best performance.

Clearly, the problem of aiming and evasion does have the desired structure. It has two hostile opponents each of whom can only perform well at the expense of the other. It also has the all important time delay between the marksman's current information and the evader's future position. The problem has very little other structure to complicate or confuse the analysis: there is no noise or error in the
marksman's measurements; the marksman has no control however indirect over the evader's motion; the evader has no destination to reach, no fuel to conserve, no mission to accomplish other than his own survival.

In this connection, a fundamental difference between this formulation and those of previous authors is that the marksman's weapon is capable of continuous uninterrupted fire. This means that the marksman need only decide where to aim at each instant. Past formulations have allowed the hapless marksman only a single bullet so that not only must he decide where to aim but also when to fire. This not only increases the marksman's computational and analytical burden (he must make two decisions instead of one at each instant of time) but also, in the case of a partie of infinite duration, introduces mathematical difficulties of such magnitude that no optimal strategy for the marksman exists! This mathematical anomaly is neatly avoided here by furnishing the marksman with a more modern weapon.

There is also a fundamental difference between the "problem of aiming and evasion" and the currently much-discussed "problem of pursuit and evasion". In the former, the marksman is stationary and his bullets unguided — his problem is purely one of prediction. Problems of pursuit and evasion, on the other hand, are characterized by a mobile, steerable pursuer who attempts to maneuver (subject to various dynamic limitations) in such a way that an intercept with the evader is attained. This difference between the two problems is usually real, i.e., their mathematical formulations are rarely homomorphic. Thus, as always, one
must be careful to apply the mathematical results obtained herein only to those problems to which they are applicable.

1.2 Aiming and Evasion and Game Theory

It should be immediately clear that this problem is under the aegis of the theory of games. This, the reader will recall, is the study of conflict situations involving two or more intelligent participants whose objectives are conflicting. The problem of aiming and evasion is certainly in this form.

It has been over a quarter of a century since John von Neumann and Oskar Morgenstern introduced the theory of games [1]. Their work has produced such essential items as the concept of a strategy, the Value of a game and, perhaps most important of all, the clear and practical delineation of an optimal strategy which might be either pure or mixed. The reader should be conversant with these concepts, at least as they apply to two-person zero-sum games. However, further technical knowledge of game theory will not be necessary for the understanding of this work. For those who wish to refresh their memory of the subject, several excellent texts [1], [2], [3], invite perusal.

In particular, the game (The sobriquet "game" may now be substituted for "problem" since the problem has been shown to be a game.) of aiming and evasion is quite clearly of the two-person variety. Also, because the motives of the marksman are homicidal while the evader desires the opposite, survival, the game appears to have strong zero-sum tendencies. This natural proclivity should be encouraged since
the two-person zero-sum game is undoubtedly the simplest possible non-trivial game. Thus the problem, when precisely stated in Chapter III, will be cast in the zero-sum, two-person mold. It should be emphasized, however, that the problem is naturally in this form — no force fit has been required.

It is important to understand at the outset that any optimal solution of the aiming and evasion game will require mixed strategies for the protagonists. The proof is by contradiction: Assume that an optimal pure strategy exists for the evader. Then the marksman, whose intelligence is assumed, can calculate it as readily as can the evader and thus he (the marksman) can achieve a unit hit probability with each shot. Since the evader can easily improve on this dismal result by randomizing, the contradiction is established. A similar argument applies for the marksman's optimal strategy.

The implication of this result is that, for the aiming and evasion game, no simple solution of deterministic type so often encountered in Modern Control Theory will be forthcoming and that the road ahead will be difficult.

1.3 Synopsis of the Dissertation

The dissertation is organized as follows. Chapter I, of which this synopsis is the last section, contains a word statement of the "problem of aiming and evasion" and a description of some of the reasons why it was chosen for study. As is explained, the problem of aiming and evasion is a conflict situation and hence is under the aegis of game
theory. Specifically, it is shown that the problem is a two-person zero-sum game and that the players' optimal strategies will necessarily be mixed.

Chapter II presents an exhaustive listing and a discussion of all of the past published work available on the problem. The main mathematical development then begins in Chapter III where a whole sequence of aiming and evasion games is defined beginning with a very general formulation and followed by several progressively simpler and more specific versions. The last and least complex in this sequence is called the Austere Game and is the focus of the dissertation. It is important however to trace and understand the whole evolution of the problem from the most general to the most simple and specific so that each simplifying assumption can be thoroughly understood, its consequences delineated, and its relationship to the original parent problem pinpointed.

Chapter IV proves the existence of solutions to these games. This follows from the finiteness of their definitions — they are characterized by a discrete, integer-valued time variable, a termination rule that assures a finite duration, and a finite set of feasible pure strategies — so that the fundamental theorem of game theory which guarantees the existence of solutions is applicable.

It is also shown in Chapter IV that it is possible, at least in principle, to convert the Austere Game to "normalized form" from the "extensive form" defined in Chapter III. However, in practice, the resulting payoff matrix has so many rows that this is impossible.
This leads rather naturally to the presentation in Chapter V of a "sub-normalized form" which has most of the theoretical and computational advantages of the normalized form yet has many orders of magnitude fewer rows.

Chapter VI presents two computational techniques for the solution of aiming and evasion games. The first is linear programming. This, it is shown, can be used to solve games in sub-normalized form in much the same way as it is used to solve games in normalized form: the solution of a linear program yields the optimal strategy for the evader, the solution of its dual gives the optimal strategy for the marksman. The second computational technique, that of dynamic programming, is derived by solving the linear programs sequentially and is shown to be quite useful for solving games of long duration.

Chapter VII details the application of graph theoretic concepts to the study of the structure of these games. The graph of the Austere Game is introduced and its properties, among them a high degree of symmetry, are discussed. It is pointed out that the number of paths through the graph is an important parameter and counting techniques for them are developed.

Many of these concepts are utilized in Chapters VIII and IX. In the former, techniques for deriving bounds on the optimal payoff are given and applied to obtain an upper and lower bound on it. These are plotted in several figures and serve to give valuable insight into the numerical magnitude of the payoff and its dependence on the
various parameters of the game. Chapter IX discusses a class of approximations to the optimal strategies. First, the general properties of this class are defined and then several of its simpler members are analyzed in detail.

Chapter X, perhaps the most important chapter of the dissertation, presents the exact analytical solution for a whole class of Austere Games. This solution, whose structure is both elegant and appealing, is discussed at length, closed forms are derived, and its behavior as the duration of the game approaches infinity is delineated.

A summary and discussion of the main results, the conclusions reached, and ideas for future research are contained in Chapter XI, the last chapter of the dissertation.
CHAPTER II
PRIOR WORK

The history of aiming and evasion problems extends back to the early fifties when game theory was young and hope and interest in it were high. The first extremely simple aiming and evasion game was apparently formulated by Rufus Isaacs and described by him at a meeting of the Operations Research Society of America on May 16, 1953. The game he suggested, which shall be called "Isaacs' Game", is perhaps the most simple non-trivial one possible and is in fact closely related to the games analyzed herein. It does, however, have infinite duration and allows the marksman but a single bullet so that, as those working on it eventually discovered, no optimal strategy for the marksman exists. They also discovered that, despite the problem's innocuous appearance, it is technically quite difficult so that, overall, probably several man-years were devoted to it.

Isaacs' description piqued the interest of L. E. Dubins who was the first to analyze the game. This appeared first in a paper with limited distribution [4], then some years later in a form available to a wider audience [5]. Simultaneously and independently a group at RAND Corporation was obtaining the same results. These were published first as internal RAND Corporation memoranda [6], [7] and later in the open literature [8], [9]. Some of the material has
also found its way into Isaacs' book [10, pages 343-4 and 355-6].

Meanwhile, others were taking a different tack. Herbert Scarf and L. S. Shapley studied general versions of games with time lags in the information of one or both participants and obtained a set of functional equations describing them. These functional equations, although (and perhaps because) they are quite general, are rather difficult to apply and are not particularly insightful. This is in contrast to the functional equations derived in Chapter VI of this dissertation which are clear and useful although not nearly so general.

Scarf and Shapley's work appeared first as an internal RAND Corporation memorandum [11] and later in the open literature [12].

A different problem and technique were described by E. W. Gröneweld [13]. His approach was to constrain the evader's motion to be sinusoidal whose amplitude and period are parameters freely chosen by the evader subject to certain constraints. The marksman's measurements are assumed to be corrupted by white noise. In addition, he is constrained to the use of a first-order extrapolator operating on smoothed observation data. With all these assumptions, Gröneweld was able to solve for the optimum parameters of the evader's motion and the optimum smoothing gains in the marksman's tracking filters.

Finally, a series of problems roughly related to the problem of aiming and evasion have been analyzed by Ho [14], Speyer [15], and Bulfer and Hemami [16], [17]. These entail a linear set of dynamics
for the evader and a quadratic payoff. However, they are not really the game of aiming and evasion because the payoff does not depend on the accuracy of the predictions, only on the accuracy of the tracking. Thus the work while interesting is irrelevant.

The material cited above represents some 18 years of work on the problem of aiming and evasion. The sum total seems disproportionately small. Most of the results have been obtained for "Isaacs' Game" which is one of the simplest non-trivial versions imaginable. Why have the results been so meager? The reasons are probably two-fold:

1. The marksman's aiming point and the evader's next move apparently must depend on the whole past history of the evader's trajectory. No one has yet found a way to do this. Neither has anyone been able to find a sufficient statistic for the entire past trajectory. Problems of this type also arise in Modern Control Theory which likewise does not have a very good solution. Chapter II of Aoki's book [18] has a good discussion of this topic.

2. Problems of the aiming and evasion genre really belong most naturally to and should be formulated as differential games [10]. However, the present state of knowledge of differential games allows only games of perfect information [10] which the game of aiming and evasion is emphatically not! At present, no one even knows a practical way of implementing mixed strategies in differential games. This, in the author's opinion, is the crippling weakness of differential game theory and will severely limit the theory's
application and usefulness until it is eliminated. As has been shown, strategies in the aiming and evasion game must be mixed which immediately precludes a differential games formulation, at least at this writing.
CHAPTER III
PROBLEM FORMULATION

This chapter is devoted to the precise mathematical formulation of several classes of aiming and evasion problems. Initially, in section 3.1, a very general formulation is developed and discussed. Following this a sequence of related simpler versions are introduced, each less complicated than its predecessors. These have the effect of simultaneously reducing both the complexity and the realism of the formulation while increasing the probability of a successful analytical assault. In section 3.2, the simplification obtained when the state equations are linear and the payoff function is "origin-insensitive" is detailed. The "Austere Game", described in section 3.3, is one with a particularly simple linear state equation and origin-insensitive payoff. It is this game which shall be the focus of this dissertation. Finally, the game analyzed by Isaacs which is closely related to a specific Austere Game is presented in section 3.4 together with its solution. A few words of discussion and conclusions, section 3.5; terminates the chapter.

3.1 General Formulation

In general, any formulation of an aiming and evasion problem must specify: 1) a dynamic model of the evader (i.e., the set of dynamic limitations on his motion); 2) a dynamic model for the marksman (if
any); 3) the information available to each; 4) a payoff function or performance criterion for each; and 5) a definition of optimality. Each of these elements will be discussed in turn below.

For several reasons, discrete time seems most convenient for this problem. One reason is that the alternative, continuous time, means a differential game formulation which as discussed in Chapter II should be avoided. It should also be mentioned that in this, the digital era, most originally continuous problems must be "discretized" anyway to enable their solution by digital computer. This means that a discrete formulation is really the most desirable form while a continuous version represents something of an inconvenience.

Thus, it shall be assumed that time is not continuous but is quantized, that (using whatever scaling is necessary) it may take on only integer values, that the problem starts at some fixed initial time 0 known to both players, and that it ends at some fixed final time $N$ also known to both. In other words, time may be represented by the integer-valued variable $k = 0, 1, 2, \ldots, N$.

The evader has under his control and must choose the sequence:

$$u_1, u_2, \ldots, u_N$$

from a feasible set $U$. These are the evader's control variables. They determine the evader's state $x_k$ at time $k$ by the difference equation:

$$x_k = f_k(x_{k-1}, u_k) \quad u_k \in U \quad k = 1, 2, \ldots, N \quad (3-1)$$

where $x_0, x_1, x_2, \ldots, x_N$ are in general $n$-vectors, $f_k$ is a vector-valued
function of $x_{k-1}$ and $u_k$, and $x_0$ is assumed known to both participants. The $n$-dimensional vector space containing $x_k$ will be denoted by $\mathbb{R}^n$. At time $k$ when the evader must choose his control variable $u_k$, it is assumed he has knowledge of his past states $x_{k-1}$, $x_{k-2}$, ..., $x_0$ and his past controls $u_{k-1}$, $u_{k-2}$, ..., $u_1$ but that he has no information whatsoever of the marksman's present or past aiming points. Thus his control must be in a sense "open-loop" in that he must make the marksman miss without ever knowing where the marksman is aiming.

At this point, a simplifying assumption will be made. It will be assumed that the set $U$ of evader's feasible controls is finite. Its cardinality is $K$, a finite integer. This assumption is not quite as restrictive as it seems. For one thing, although $K$ must be finite it can be very, very large, large enough to approximate infinity (in an engineering sense). In other words, if necessary $K$ can always be made large enough to yield a satisfactory approximation to any practical situation. A second justification harks back to the assumption that a digital computer will ultimately be used to calculate the optimal strategies. Since present-day digital computers have finite memory and finite word length, a finite set of feasible controls is a necessity.

Since $f_k$ may be any function, no generality is lost by restricting $U$ to be the integers from 0 to $K-1$:

$$U = \{0, 1, 2, \ldots, K-1\}. \quad (3-2)$$
The marksman has under his control and must choose a sequence of n-vectors: \( \hat{x}_{\lambda}, \hat{x}_{\lambda+1}, \ldots, \hat{x}_N \). These are the marksman's aiming points, i.e., \( \hat{x}_k \) is the position of the bullet at the end of its flight. The bullet's time delay \( \lambda \) is an integer known to both participants. In other words, a bullet fired at time \( k-\lambda \) reaches its destination \( \hat{x}_k \) at time \( k \). It is important to note that the subscript on \( \hat{x}_k \) denotes the arrival time of the bullet, not the time at which the bullet is fired.

The aiming point \( \hat{x}_k \) should be the marksman's best estimate of \( x_k \); i.e., his best performance is achieved when \( \hat{x}_k = x_k \), a "direct hit" at time \( k \).

The marksman may choose \( \hat{x}_k \) with no dynamic restrictions whatsoever and with full knowledge of all past aiming points: \( \hat{x}_{k-1}, \hat{x}_{k-2}, \ldots, \hat{x}_\lambda \) and of target states: \( x_{k-\lambda}, x_{k-\lambda-1}, \ldots, x_0 \).

The payoff to the marksman which he desires to maximize, is restricted to the form:

\[
J = E \left[ \sum_{k=\lambda}^{N} h_k(x_k, \hat{x}_k) \right] \tag{3-3}
\]

where the expectation operator \( E \) is required since the strategies are expected to be mixed which means that \( x_k \) and \( \hat{x}_k \) will normally be random variables. In this equation, \( h_k \) is a real-valued function which is intended to be a quantitative measure of the effectiveness of the \( k \)th shot. In other words, \( h_k(x_k, \hat{x}_k) \) is the degree of accuracy if the marksman's bullet arrives at \( \hat{x}_k \) at time \( k \) while the evader's actual state is \( x_k \). It should be understood that equation (3-3) represents a rather important assumption, to wit, the total payoff to
the marksman as a result of his $S = N - \lambda + 1$ shots may be expressed as the superposition of the payoffs from each individual one.

It is not as restrictive as it sounds however. There are a number of practical and interesting problems which do indeed have this type of payoff. An example is the case in which the payoff is the average aiming error (averaged over all $S$ shots):

$$ J = -\frac{1}{S} \sum_{k=\lambda}^{N} \left| |x_k - \bar{x}_k| \right| $$

or the mean-square aiming error:

$$ J = -\frac{1}{S} \sum_{k=\lambda}^{N} \left| |x_k - \bar{x}_k| \right|^2. $$

In these equations, $J$ represents the payoff to the marksman which he desires to maximize. Naturally, he does not want to maximize the average or mean-square aiming error! He would rather minimize it, which is equivalent to maximizing its negative, and explains the appearance of the minus sign. The norm of a vector $v$, written $||v||$, is a measure of its "length" or "magnitude" and may be any of the commonly defined norms. The Euclidian norm, for example, is:

$$ ||v|| = \sqrt{(v^1)^2 + (v^2)^2 + \ldots + (v^n)^2} $$

where $v^1$, $v^2$, ..., $v^n$ are the components of the vector $v$.

A payoff function which apparently is not in the class given by equation (3-3) is the overall hit probability:

$$ J = 1 - \prod_{k=\lambda}^{N} (1-p_k) $$
where \( p_k \) is the probability of a hit at time \( k \). This equation can be expanded to yield:

\[
J = \sum_{k=1}^{N} p_k + \text{higher order terms in } p_k.
\]

If the individual \( p_k \) and their sum \( J \) are small, the higher order terms may be neglected. Now, if a hit function \( H_k(x_k, \hat{x}_k) \) is defined to be 1 if \((x_k, \hat{x}_k)\) represents a "hit" and 0 otherwise, then \( J \) may be rewritten:

\[
J = \mathbb{E}\left[ \sum_{k=1}^{N} H_k(x_k, \hat{x}_k) \right]
\]  

(3-4)

the expected number of hits. The approximation sign is used here to denote the negligence of the higher order terms and will henceforth be dropped, i.e., the expected number of hits will henceforth be considered a legitimate payoff function in its own right, which of course it is since it is in the form of equation (3-3).

The value of \( J \) given in equation (3-3) represents the payoff to the marksman — what then is the evader's payoff? As has been pointed out, for several reasons the game fits most naturally into the zero-sum format. Therefore it shall be assumed that the payoff to the evader which he desires to maximize is \(-J\). This means that whatever is paid to the marksman is forfeited by the evader and vice versa. The marksman desires to maximize \( J \) his payoff. Conversely, the evader must attempt to maximize his own payoff \(-J\), that is, the evader attempts to minimize \( J \).
The zero-sum assumption leads to a straightforward definition of optimality. Let the symbol \( \phi \) denote the rule or algorithm by means of which the evader chooses his controls \( u_1, u_2, \ldots, u_N \). Let \( \psi \) denote the rule or algorithm by which the marksman, using all information available to him at the time, calculates his aiming points \( \hat{x}_\lambda, \hat{x}_{\lambda+1}, \ldots, \hat{x}_N \). These algorithms may and indeed probably will be quite complicated. Nevertheless each player, since he must make the decisions somehow, will use some computational procedure to make his choice and it is these computational procedures which are symbolized by \( \phi \) and \( \psi \).

In game theoretic parlance, \( \phi \) and \( \psi \) are called "strategies". Once they are chosen by the evader and marksman respectively the payoff, \( J \), is determined, a functional dependence which will be denoted here by writing \( J(\phi, \psi) \). As von Neumann and Morgenstern [1] have shown, a reasonable definition of the optimal strategies \( \phi^* \) and \( \psi^* \), is that they correspond to a saddle-point of \( J \). A saddle-point, when it exists, is defined to be the pair \( (\phi^*, \psi^*) \) such that:

\[
J(\phi^*, \psi) \leq J(\phi^*, \psi^*) \leq J(\phi, \psi^*)
\]

for all \( \phi, \psi \). The scalar constant \( V = J(\phi^*, \psi^*) \) is called the "Value" of the game.

The full justification for this definition of optimality is given in reference [1]. Briefly, however, it is based on the following arguments:
1. Generally it will not be possible to globally maximize $J$ with respect to $\psi$ and simultaneously globally minimize $J$ with respect to $\phi$. A weaker condition must be found.

2. If a saddle-point exists, then from the marksman's point of view, there is a lower bound $V$ on the payoff. More precisely, there exists a strategy $\psi^*$ for the marksman such that his payoff is at least $V = J(\phi^*, \psi^*)$ no matter what the evader does. It might be higher, but he is guaranteed at least this much. Mathematically, this lower bound is written:

$$V = J(\phi^*, \psi^*) \leq J(\phi, \psi^*).$$

3. If a saddle-point exists, then from the point of view of the evader there is an upper bound on the payoff. There exists a strategy $\phi^*$ for the evader such that $J$ is no more than $V = J(\phi^*, \psi^*)$ no matter what the marksman does. It might be lower (the evader being the minimizing player desires $J$ as low as possible) but he is guaranteed that it will be no greater than $V$. Mathematically, this upper bound is written:

$$J(\phi^*, \psi) \leq J(\phi^*, \psi^*) = V.$$

4. Since the marksman can make the payoff at least $V$ and the evader can prevent it from being more than $V$, $V$ is a logical candidate for the optimal payoff and $\phi^*$ and $\psi^*$ the optimal strategies.

All of this assumes that a saddle-point exists. Chapter IV will address itself to the question of the existence of saddle-points for the
games of interest to this paper. If a saddle-point does exist then an equivalent definition of it is:

$$V = \min_{\phi} \max_{\psi} J(\phi, \psi) = \max_{\psi} \min_{\phi} J(\phi, \psi). \quad (3-6)$$

If optimal strategies do not exist but a value does, then equation (3-6) should be replaced by:

$$V = \inf_{\phi} \sup_{\psi} J(\phi, \psi) = \sup_{\psi} \inf_{\phi} J(\phi, \psi).$$

In this case, optimal strategies $\phi^*$ and $\psi^*$ might not exist but a pair of "ε-optimal strategies" $(\phi_\varepsilon, \psi_\varepsilon)$ defined by:

- $J(\phi_\varepsilon, \psi) \leq V + \varepsilon$
- $J(\phi, \psi_\varepsilon) \geq V - \varepsilon$

for all $\phi, \psi$ and $\varepsilon > 0$ exist and can be used.

So concludes a statement of a very general class of aiming and evasion problems. In the following pages a sequence of simplifications of it will be detailed. To aid the cause of pedagogical clarity, this original version shall be referred to as the "Protoproblem" to distinguish it from those which follow.

3.2 The Case of Linear Equations and Origin-Independent Payoff

If the state equation (3-1) is linear and the payoff function (3-3) is independent of the origin of $\mathbb{R}^n$, the $n$-dimensional vector space containing $x_k$ and $\dot{x}_k$, a certain degree of simplification results. Suppose that the state equation is:
\[ x_k = A x_{k-1} + b(u_k); \quad x_0 = 0; \quad k = 1, 2, \ldots, N \]  

(3-7)

where \( A \) is a constant \( n \times n \) matrix (the extension to the case of \( A \) and \( b \) varying with \( k \) is a trivial generalization which will not be considered here) and \( b \) a vector-valued function of \( u_k \). Further, suppose that the functions \( h_k \) in the payoff equation satisfy:

\[ h_k(x_k, \tilde{x}_k) = h_k(x_k - \alpha, \tilde{x}_k - \alpha) \]  

(3-8)

for all \( \alpha \in \mathbb{R}^n \). Then, it is possible to make the change of variables:

\[ z_k = x_k - A^{\frac{\lambda}{\lambda-1}} x_{k-\lambda} = b(u_k) + Ab(u_{k-1}) + \ldots + A^{\frac{\lambda-1}{\lambda-1}} b(u_{k-\lambda+1}) \]  

(3-9)

\[ \tilde{z}_k = \tilde{x}_k - A^{\frac{\lambda}{\lambda-1}} x_{k-\lambda} \]

and with these substitutes, the Protoproblem may be rewritten as:

**The CLEOP Problem:**

(Case of Linear Equations and Origin-independent Payoff)

Find the \( K \)-ary, \( N \)-tuple \( u^* = (u_1^*, u_2^*, \ldots, u_N^*) \in U^N \) and an algorithm \( \psi^* \) for calculating a sequence of aiming points \( \tilde{z}_k = (\tilde{z}_k^*, \tilde{z}_{k+1}^*, \ldots, \tilde{z}_N^*) \) which yield a saddle-point of the payoff function:

\[ J(u, \psi) = E\left[ \sum_{k=\lambda}^{N} h_k(z_k, \tilde{z}_k) \right] \]

\[ = E\left[ \sum_{k=\lambda}^{N} h_k(b(u_k) + Ab(u_{k-1}) + \ldots + A^{\frac{\lambda-1}{\lambda-1}} b(u_{k-\lambda+1}), \tilde{z}_k) \right]. \]  

(3-10)

The CLEOP Problem is clearly equivalent to the Protoproblem when \( (3-7) \) and \( (3-8) \) hold since:
1. Equation (3-8) guarantees the payoffs are equal.

2. At time $k$, $x_{k-\lambda}$ is known to each participant so that knowledge or lack thereof of $x_k$ is equivalent to knowledge or lack thereof of $x_k$. The transformation (3-9) is simply a time-varying shift of the origin.

3. By stating the CLEOP Problem in this way emphasis is placed on the fact that, since the evader gets no information during a partie, it makes no difference whether he uses an open-loop strategy $u = (u_1, u_2, \ldots, u_N)$ or a closed-loop strategy $\phi$. The two are equivalent in that they yield the same payoff.

Sometimes, especially when $n$ is large, it is useful to reformulate the problem as a $\lambda$-dimensional one. This transformation often results in a considerable dimensionality reduction. However, even when $\lambda \geq n$ so that no reduction in dimensionality is possible, the reformulation can add insight into the structure of the game and into the crucial role of the parameter $\lambda$.

To do this, one must redefine the "state" as the vector $w_k = [u_k, u_{k-1}, \ldots, u_{k-\lambda+1}]^T$. Here, $w_k$ is a $K$-ary $\lambda$-vector: $w_k \in U^\lambda$. Then, one can rewrite the CLEOP Problem as:

State equation: $w_k = F w_{k-1} + G u_k$

Constraint: $u_k \in U$

Payoff: $J = E\left[\sum_{k=\lambda}^{N} g_k(w_k, z_k)\right]$

where:
\[
\begin{bmatrix}
0 & 0 & 0 & \ldots & 0 & 0 \\
1 & 0 & 0 & \ldots & 0 & 0 \\
0 & 1 & 0 & \ldots & 0 & 0 \\
0 & 0 & 1 & \ldots & 0 & 0 \\
0 & 0 & 0 & \ldots & 1 & 0 \\
0 & 0 & 0 & \ldots & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}
\begin{bmatrix}
1 \\
w_1^k \\
w_2^k \\
w_3^k \\
w_4^k \\
w_\lambda^k
\end{bmatrix}
\]

and:
\[
g_k(w_k, \hat{x}_k) = h_k(b(w_k^1) + Ab(w_k^2) + A^{2}(w_k^3) + \ldots + A^{\lambda-1}b(w_k^\lambda), \hat{x}_k).
\]

Thus, the dimensionality of this form of the CLEOP Problem is not equal to \( n \) the dimensionality of the original state space \( \mathbb{R}^n \) but instead is equal to \( \lambda \) the time delay. The value of \( \lambda \) is therefore truly a fundamental parameter of the system.

This section concludes with an anticlimactic but important result. The function \( h_k(x_k, \hat{x}_k) \) will have the origin-independent property (3-8) if and only if it may be written as some function \( H_k \) of the error vector \( x_k - \hat{x}_k \):

\[
h_k(x_k, \hat{x}_k) = H_k(x_k - \hat{x}_k).
\]

The proof is obtained by simply letting \( \alpha \) in (3-8) be equal to \( \hat{x}_k \) yielding:
Since the aiming error is, in the final analysis, the measure of the goodness of an aiming system, this result practically guarantees that most payoff functions will have the origin-independent property.

3.3 The Austere Problem

Since it is desirable to have a concrete problem for the dissertation to assault, and since it is a stated aim to eliminate all unnecessary embellishments, the following, perhaps the simplest subproblem in the class of CLEOP Problems, is defined:

The Austere Problem:

The member of the class of CLEOP Problems for which:

\[ n = 1 \quad A = 1 \quad b(u_k) = u_k \]

and:

\[ h_k(x_k, \hat{x}_k) = \begin{cases} 1 & \text{if } x_k - \hat{x}_k = 0 \\ 0 & \text{otherwise} \end{cases} \]

In terms of the Protoproblem, this definition becomes:

The Austere Problem:

The state of the evader at time \( k \) is represented by the scalar-valued variable \( x_k \) related by the state-equation:

\[ x_k = x_{k-1} + u_k; \quad x_0 = 0; \quad k = 1, 2, \ldots, N. \quad (3-11) \]
The evader's controls are constrained to be elements of the finite feasible set $U$:

$$u_k \in U = \{0, 1, 2, \ldots, K-1\}.$$  \hfill (3-12)

The marksman's aiming points are the scalars $\hat{x}_\lambda, \hat{x}_{\lambda+1}, \ldots, \hat{x}_N$.

The payoff is the expected total number of "direct hits":

$$J = E[\sum_{k=\lambda}^N h(x_k - \hat{x}_k)].$$  \hfill (3-13)

A "direct hit" is defined as the case where the aiming point exactly equals the evader's state: $\hat{x}_k = x_k$ so that $h$ is defined:

$$h(y) = \begin{cases} 
1 & \text{if } y = 0 \\
0 & \text{otherwise.}
\end{cases}$$  \hfill (3-14)

The problem is to find a saddle-point of $J$ in the strategies $\phi$ and $\psi$. Since for this problem the case $\lambda = 1$ is trivial, it will be assumed in the sequel that $\lambda$ is a positive integer greater than one.

It is now convenient to specify more precisely what the strategies $\phi$ and $\psi$ really are. $\phi$ is the set of mappings $\phi = \{\phi_1, \phi_2, \ldots, \phi_N\}$ such that:

$$u_1 = \phi_1$$
$$u_2 = \phi_2(u_1)$$
$$u_3 = \phi_3(u_1, u_2)$$  \hfill (3-15)

$$\quad \ldots$$
\[ u_k = \phi_k(u_1, u_2, \ldots, u_{k-1}) \]

\[ u_N = \phi_N(u_1, u_2, \ldots, u_k, \ldots, u_{N-1}) \]

while \( \psi = \{\psi_1, \psi_{i+1}, \ldots, \psi_N\} \) where:

\[ \hat{x}_\lambda = x_0 + \psi_\lambda \]

\[ \hat{x}_{\lambda+1} = x_1 + \psi_{\lambda+1}(u_1) \]

\[ \hat{x}_{\lambda+2} = x_2 + \psi_{\lambda+2}(u_1, u_2) \]

\[ \hat{x}_k = x_{k-\lambda} + \psi_k(u_1, u_2, \ldots, u_{k-\lambda}) \]

\[ \hat{x}_N = x_{N-\lambda} + \psi_N(u_1, u_2, \ldots, u_k, \ldots, u_{N-\lambda}). \]

The reader should note the following points about the Austere Problem:

1. In defining the strategies \( \phi \) and \( \psi \), advantage has been taken of the fact that, since \( x_0 = 0 \), knowledge of \( x_1, x_2, \ldots, x_k \) is equivalent to knowledge of \( u_1, u_2, \ldots, u_k \). Thus, although at time \( k \) the evader has complete knowledge of both \( u_1, u_2, \ldots, u_{k-1} \)
and \( x_0, x_1, \ldots, x_{k-1} \), only the former need be used to calculate the next control \( u_k \). A similar argument applies to the marksman.

2. The change of variable in equation (3-9) has been surreptitiously made so that:

\[
\psi_k(u_1, u_2, \ldots, u_{k-\lambda}) = \hat{\lambda}_k - x_{k-\lambda} = \hat{\lambda}_k. \tag{3-17}
\]

3. A completely analogous argument to that of point 1 above justifies the fact that past values of the aiming point \( \hat{\lambda}_k, \hat{\lambda}_{k+1}, \ldots, \hat{\lambda}_{k-1} \) do not enter into the marksman's calculation of \( \hat{\lambda}_k \). A perhaps more intuitive argument goes as follows: Since the "bullets are invisible", i.e., the evader has no knowledge of the marksman's aiming points, they can have no effect on the evader's trajectory. This is reflected by the fact that \( \hat{\lambda}_k, \hat{\lambda}_{k+1}, \ldots, \hat{\lambda}_{k-1} \) do not appear as arguments of the evader's strategy \( \psi_k \) in equations (3-15). Since the marksman's problem is to predict the evader's future position and since this future position does not depend upon the marksman's past aiming points, then it follows that the marksman's predictions need not depend upon his past aiming points either.

4. Since from (3-11) \( x_k \) may be written:

\[
x_k = u_1 + u_2 + \ldots + u_k = \sum_{i=1}^{k} u_i \tag{3-18}
\]

then from (3-12) it follows that each of the \( x_k \) are non-negative integers:

\[
x_k \in \{0, 1, 2, \ldots, k(K-1)\}.
\]
The $z_k = x_k - x_{k-\lambda}$ are also non-negative integers:

$$z_k \in Z = \{0, 1, 2, \ldots, \lambda(K-1)\}.$$ 

This set $Z$ and its cardinality $L = \lambda(K-1) + 1$ are used so frequently in the sequel that they have been given their own names. The variable $z_k = x_k - x_{k-\lambda}$ is the difference between the evader's position at time $k$ and the marksman's last known piece of data $x_{k-\lambda}$. The variable $x_{k-\lambda}$ thus represents what the marksman knows while $z_k$ represents what he doesn't know about the evader's position $x_k$. The set $Z$, the set of all possible $z_k$, has $L$ elements. Thus, $L$ is the number of possible locations at which the evader can be after the marksman is told $x_{k-\lambda}$ — it is the marksman's remaining uncertainty in the sense that, after he is told $x_{k-\lambda}$, he knows that the evader's state $x_k$ must be one of only $L$ possibilities. With respect to $x_{k-\lambda}$, these are the $L$ elements of $Z$.

5. It follows from the preceding point that if the marksman does not want to waste any shots with guaranteed misses, he should choose his estimate $\hat{z}_k$ from $Z$:

$$\hat{z}_k \in Z = \{0, 1, 2, \ldots, \lambda(K-1)\} \quad (3-19)$$

that is, $\hat{z}_k$ should not be just anything but should be taken from the first $L$ non-negative integers. More rigorously, it can easily be shown that the restriction of equation (3-19) will never decrease the payoff $J$. 

6. The strategies φ and ψ as they are defined in equations (3-15) and (3-16) are pure, closed-loop strategies. They are "pure" (as opposed to mixed) because they are simply deterministic functions, there is no element of randomness about them. They are "closed-loop" because each control $u_k$ and estimate $\hat{z}_k$ is a function of past states.

7. It is possible and indeed at times useful to restate the evader's pure strategy as an open-loop one. This can be done by substituting the equations (3-15) one into another to yield:

$$
\begin{align*}
    u_1 &= \phi_1 \\
    u_2 &= \phi_2(\phi_1) \\
    u_3 &= \phi_3(\phi_1, \phi_2(\phi_1)) \\
    &\vdots \\
    &\vdots \\
    &\vdots 
\end{align*}
$$

Thus, a different way of specifying the evader's pure strategy is to simply specify the K-ary N-tuple $u = (u_1, u_2, \ldots, u_N)$, $u \in U^N$. What is happening is that since the evader gets no information of any value during his flight, he might just as well plan his whole trajectory ahead of time by deciding in advance which control he will use at each time. Since the set $U^N$ of all possible pure, open-loop evader's strategies has far fewer elements than the set $\phi$ of all pure closed-loop strategies, a considerable dimensionality
reduction is often obtained by specifying \( u \) rather than \( \psi \). No such reduction seems possible for the marksman.

8. It has been pointed out that mixed strategies, not pure ones, will be required of the players. This is done as follows. Each participant forms the set of all his possible pure strategies. For the marksman, this is the set \( \Psi \) of all possible \( \psi \) — usually a set of monstrous proportions. In the evader's case, it is the set \( \Phi \) of all possible \( \phi \) if closed-loop strategies are desired or the set \( U^N \) of all possible \( K \)-ary \( N \)-tuples if open-loop ones are called for. Then each participant defines a probability distribution on his set of pure strategies and, when it comes time to play the game, chooses one element of the set by means of a random experiment according to the defined probability distributions. As an example, consider the case of \( K = 2, N = 2 \) where the evader desires an open-loop mixed strategy. His set of all possible pure, open-loop strategies is the set of all ordered binary pairs:

\[
U^2 = \{00, 01, 10, 11\}.
\]

A mixed strategy is then a probability distribution defined on \( U^2 \), for example:

\[
\begin{align*}
p(00) &= 3/8 & p(10) &= 1/4 \\
p(01) &= 1/8 & p(11) &= 1/4.
\end{align*}
\]

When it comes time to play the game, the evader performs a random
experiment whose 4 possible outcomes have these probabilities. If the result is, say 10, the evader then chooses his controls to be:

\[ u_1 = 1 \quad u_2 = 0 \]

yielding the trajectory:

\[ x_0 = 0 \quad x_1 = 1 \quad x_2 = 1. \]

It is instructive to rewrite the Austere Problem in CLEOP Problem form. This yields:

**The Austere Problem:**

Find the K-ary, N-tuple \( u^* = (u_1^*, u_2^*, \ldots, u_N^*) \)

and an algorithm \( \psi^* \) for calculating a sequence of aiming points

\[ z^* = (z_{\lambda}^*, z_{\lambda+1}^*, \ldots, z_{N}^*) \in \mathbb{Z}^S \] which yields a saddle-point of the payoff function:

\[
J = E[ \sum_{k=\lambda}^{N} h(u_k + u_{k-1} + \ldots + u_{k-\lambda+1} - z_k^*) ] \tag{3-20}
\]

where:

\[
h(x) = \begin{cases} 
1 & \text{if } x = 0 \\
0 & \text{otherwise.}
\end{cases}
\]

This formulation emphasizes that the marksman’s job is to estimate a simple sum \( z_k = u_{k+\lambda+1} + u_{k+\lambda+2} + \ldots + u_k \) of the evader’s last \( \lambda \) controls.
Clearly each $z_k$ is highly correlated with the former ones, a fact that does not bode well for the evader's survival and indeed is what makes the Austere Problem non-trivial.

3.4 Isaacs' Game

It must be emphasized at the very outset that although the problem proposed and discussed by Rufus Isaacs (cf. the discussion in Chapter II) looks convincingly like the Austere Problem with $K = \lambda = 2$ and $N \rightarrow \infty$, it is not only not the Austere Problem, it is not even the Protoproblem. The reason is that instead of allowing the marksman $S = N - \lambda + 1$ shots, Isaacs allows him only one. The unlucky marksman must thus not only decide on his aiming point, he must also decide when to expend his lone bullet. This difference exerts a subtle effect. Although one might think that the marksman might be able to partition his problem into two sub-problems --- the first to choose the best aiming point at each time $k$, the second to determine the optimum $k$ at which to shoot --- and then apply the results derived herein to the first sub-problem, the fact is that although such a partition is possible, the results of this paper are not applicable to it.

Isaacs' Game:

The evader's state at time $k$ is the scalar $x_k$ determined by the scalar equation:

$$x_k = x_{k-1} + u_k; \quad x_0 = 0; \quad k = 1, 2, \ldots, N$$
where $u_k$ must be either 0 or 1:

$$u_k \in U = \{0, 1\} \quad (i.e. \ K = 2).$$

The marksman receives information on the evader's state with a delay of 2 time units (i.e., $\lambda = 2$). At some value of time $k = m$ he decides to shoot, at which point he must choose an aiming point $\hat{x}_m$ based on knowledge of $x_0, x_1, x_2, \ldots, x_{m-2}$. The payoff is the probability of a direct hit:

$$J = \Pr\{\hat{x}_m = x_m\}.$$

Here and in the pages that follow, the nomenclature $\Pr\{A\}$ should be read as "the probability of the event $A$" while the symbol $\Pr\{A|B\}$ means "the conditional probability of the event $A$ given that event $B$ has occurred".

The solution to Isaacs' Game has been obtained by Isaacs and others (see Chapter II) for the case where $N \to \infty$. It is as follows:

1. The Value $V$ of Isaacs' Game is:

$$V = \frac{3 - \sqrt{5}}{2} \approx .382.$$  \hspace{1cm} (3-21)

2. The optimal closed-loop strategy for the evader is a mixed strategy characterized by the probabilities:

$$\Pr\{u_1 = 0\} = \Pr\{u_1 = 1\} = \frac{1}{2}$$  \hspace{1cm} (3-22a)

$$\Pr\{u_k = u_{k-1}\} = 1 - V \approx .618$$  \hspace{1cm} (3-22b)
\[ \Pr\{u_k \neq u_{k-1}\} = V \approx 0.382 \]  \hspace{1cm} (3-22c)

3. No optimal strategy exists for the marksman. He should never fire
but always wait longer before shooting.

Again let it be emphasized that the Value and mixed strategy above are
not the hit probability and optimal strategy for the Austere Problem in
which \( K = \lambda = 2 \) and \( N \to \infty \). It will be shown later that by use of a
different mixed strategy the evader can improve on this performance.

3.5 Conclusions

In this chapter, a sequence of progressively simpler aiming and
evasion games has been defined. First, a very general version charac-
terized by arbitrary state equations and arbitrary functions \( h_k \) in the
additive payoff was given. Some of its features were: a discrete time
assumption, a finite set \( U \) of feasible controls, fixed initial and termi-
nal times, and a saddle-point definition of optimality. Next, in sec-
tion 3.2, attention was focused on the particular class of these games
for which the state equations are linear and the payoff is independent
of the choice of the origin of the coordinate system. For this class, a
time-varying shift of origin proved helpful and a reformulation as a
\( \lambda \)-dimensional problem was possible. In section 3.3, a specific version
of these games, called the Austere Game, in which the linear state
equations and the additive, origin-insensitive payoff were particularly
simple was presented. This is characterized by a scalar state equation
and a payoff which is the expected number of direct hits. This game, it
was pointed out, is to be the focus of analysis in the following pages. Finally, Isaacs' Game, which is roughly related to the Austere Game was presented and Isaacs' solution of it described.

The objective of this chapter and the purpose of its structure was not only to define the class of games to be studied but also to delineate and understand the whole evolution of the problem, from the most general version to the most specific, so that the role of each simplifying assumption is clarified and the results, when they are finally obtained, can be properly interpreted.
CHAPTER IV
EXISTENCE OF SOLUTIONS

It was pointed out in Chapter III that the optimal solution to these games is a saddle-point of the payoff function \( J \). Before setting out on the arduous road towards such an optimal solution, the prudent man will ask: "Is there a saddle-point?", "Does a solution in the sense that has been defined in Chapter III exist or not?" This chapter is devoted to answering these questions.

Briefly, the answer is: "Yes, the optimal solutions in terms of mixed strategies do exist!" All of the games defined in Chapter III are in extensive form. As von Neumann and Morgenstern [1] have shown, such games can always be put into normalized form since they are characterized by a finite number of choices at each time point and a termination rule that assures that the length of the game will always be finite. Section 4.1 gives the actual details of the conversion of the Austere Game into normalized form. Once it has been put into normalized form, the two-person zero-sum nature of the game immediately guarantees the existence of an optimal solution, at least in mixed strategies. A discussion of the reasoning is contained in section 4.2. A section of conclusions ends the chapter.
4.1 Conversion to Normalized Form

The normalized form of a game is generated by first making a complete listing of all possible pure strategies for the evader and a similar one for the marksman. The normalized form is a list specifying the payoff for each possible combination of one strategy for the evader and one strategy for the marksman.

Mathematically, if the set of all possible marksman's strategies is $\Psi$, the set of all possible evader's strategies is $U^N$, (Throughout this chapter, it will be assumed that open-loop evader's strategies are desired.) and the payoff for a particular strategy pair $(u,\psi)$ is:

$$J(u,\psi) \quad u \in U^N \quad \psi \in \Psi$$

then the normalized form of the game is the set $\Omega$:

$$\Omega = \{J(u,\psi); \quad (u,\psi) \in U^N \times \Psi\} \quad (4-1)$$

the set of the payoffs accruing to each possible strategy pair.

Typically, this set is presented as a two-dimensional array or matrix in which each column represents a particular strategy for the evader, i.e., an element of $U^N$, and each row represents a particular strategy for the marksman, i.e., an element of $\Psi$. There is a column for each element of $U^N$ and a row for each element of $\Psi$. The number written at the intersection of a particular row and column is the payoff for that particular pair of strategies. This matrix of numbers is called the "payoff matrix" which will be denoted by $P$. 
In the case of the Austere Problem, there are $K^N$ elements in $U^N$ and thus there are $K^N$ columns in the payoff matrix. This is usually a rather large but conceivable number.

The calculation of the number of rows in $P$ is only slightly more difficult. Each element $\psi$ of $\Psi$ is itself a set:

$$\psi = \{\psi_1, \psi_2, \ldots, \psi_N\}$$

containing a total of $S = N - \lambda + 1$ elements. Each element $\psi_k$ of $\psi$ is a function:

$$\psi_k: U^{k-\lambda} \rightarrow Z \quad (4-2)$$

(cf. equations (3-17) and (3-19)). Note that both the domain and the range of $\psi_k$ are finite sets: $U^{k-\lambda}$ has $K^{k-\lambda}$ elements and $Z$ has $L = \lambda(K-1) + 1$ elements. Thus, there will be:

$$M_k = L^{K^{k-\lambda}} \quad k = \lambda, \lambda+1, \ldots, N \quad (4-3)$$

different possible $\psi_k$ and $M$ elements in $\Psi$:

$$M = \prod_{k=\lambda}^{N} M_k = \prod_{i=0}^{S-1} L^{K^{i}} \quad (4-4)$$

This is the number of rows in $P$. In most cases of interest, $M$ is a staggeringly large number. For example, a rather small Austere Game is one in which $K = 2$, $\lambda = 2$, and $N = 8$. For this case, the payoff matrix $P$ has 256 columns and approximately $3.93 \times 10^{60}$ rows!
Now $3.93 \times 10^{60}$ is a number which commands respect! Suppose for example that using pencil and paper it takes one second to write each row. Since there are 31,536,000 seconds in a year (31,622,400 in a leap year), it would take approximately $1.3 \times 10^{53}$ years to write all of $P$. Since the universe is probably only about 10 billion years old, if one had started writing $P$ during the "big bang" of creation, he would only have managed to write about $10^{-41}$ percent of it by now. Moreover, since there are only about $3 \times 10^{58}$ milligrams of mass in the entire universe, the writer must be careful to use no more than 10 millionths of a gram of pencil lead and paper for each row or he will run out before completing his task.

Despite these comparisons, the main point is that $P$, while admittedly large, is finite and thus, in principle, is no different than a $2 \times 2$ or any other finite matrix.

4.2 Proof of Existence

Since $P$ is a finite matrix, von Neumann's fundamental theorem, which states that a saddle-point at least in mixed strategies always exists in games with finite payoff matrices, is applicable. This very general theorem, is the result upon which the whole of two-person zero-sum game theory is based. It was first proven by von Neumann over 40 years ago [20] and has since been proven by many different authors in many different ways. A fairly elementary proof is given in reference [1].
4.3 Conclusions

In this chapter it has been shown how to convert the Austere Game from the extensive form in which it was formulated in Chapter III into normalized form. It is important to be able to do this since both the proof of the existence of solutions and, as will be seen later, known computational techniques are dependent upon the game being in this form. Once the Austere Game was transformed into normalized form, the existence of a solution followed directly from von Neumann's fundamental theorem on two-person zero-sum games.

Also in this chapter, the size of the normalized form was calculated and the magnitude of the task of simply writing it down was estimated. Unfortunately, this turned out to be almost inconceivably large. Thus any numerical techniques for solving games which depend on the game being in normalized form will apparently not be usable, at least not without modification.
CHAPTER V
THE SUB-NORMALIZED FORM

It is unfortunate that the normalized form of the Austere Game cannot be completely written down since many known techniques for the solution of games are based upon this format. In this chapter a "not-quite-so-normalized" form called the sub-normalized form will be developed. This form has most of the notational and computational advantages of the normalized form without being quite so lengthy.

The computational advantages of the sub-normalized form deserve particular emphasis. Many known numerical techniques for solving games can only be applied to games in normalized form. If the normalized form has $10^{60}$ rows it obviously is impractical to attempt a numerical solution. However, as will be shown in Chapter VI, a few simple modifications allow the same numerical techniques to work for games in sub-normalized form as well and since the sub-normalized form is a more realistic size the computational approach becomes feasible.

Before the sub-normalized form is presented, an intermediate result, the semi-normalized form must be developed. This is done in section 5.1 and illustrated by an example in section 5.2. The sub-normalized form is then derived in section 5.3 and exemplified in section 5.4. A discussion, section 5.5, concludes the chapter.
5.1 The Semi-Normalized Form

In the normalized form of the game, the marksman must choose one strategy $\psi$ from an inconceivably large sized set $\Psi$ of possible strategies. As was shown, $\psi$ is itself a set:

$$\psi = \{\psi_1, \psi_{\lambda+1}, \ldots, \psi_N\}$$

containing $S = N - \lambda + 1$ elements. Part of the reason that $\Psi$ is so large and thus the number of rows of $P$ is so large is that there are many different possibilities for each of the $\psi_k$, so that the number of different combinations of all of the $\psi_k$ is overwhelming!

This is convincingly illustrated in equations (4-3) and (4-4). The number $M_k$ of different possible choices of $\psi_k$ in equation (4-3) is large enough in itself but the number $M$ of possible combinations of these in equation (4-4) is frightening! It is this latter equation which in great part is the culprit. If only that were a sum instead of a product ...

It is important to remember that the marksman's choices of $\psi_k = \lambda, \lambda+1, \ldots, N$ have no effect on the evader's decisions because the evader receives no information about what the marksman's choices are. Since the marksman is attempting to predict the evader's future position and since this is independent of the marksman's past predictions, it follows that the marksman's own decisions are independent of his past predictions.
This suggests that rather than make a single choice \( \psi \) from a tremendously large set \( \Psi \) of possible choices, the marksman could instead make a sequence of decisions, \( \psi, \psi_{\lambda+1}, \ldots, \psi_{N} \), i.e., he could choose one \( \psi_{k} \) at a time each from a smaller set \( \Psi_{k} \) of possibilities.

This does not really alter the dimensions of the marksman's problem. There are \( M_{k} \) (given by equation (4-3)) elements in \( \Psi_{k} \) and there are still a total of \( M \) (equation (4-4)) different possible combinations. It does offer however the possibility of writing everything down. A further aid to this possibility is the additive nature of the payoff:

\[
J = \sum_{k=\lambda}^{N} h(u_{k} + u_{k-1} + \ldots + u_{k-\lambda+1} - \psi_{k}(u_{1}, u_{2}, \ldots, u_{k-\lambda}))
\]

This new, less lengthy form of the game shall be called the semi-normalized form. It is defined by the set \( \Omega' \):

\[
\Omega' = \bigcup_{k=\lambda}^{N} \omega_{k} = \omega_{\lambda} \cup \omega_{\lambda+1} \cup \ldots \cup \omega_{N}
\]

where:

\[
\omega_{k} = \{h(u_{k} + u_{k-1} + \ldots + u_{k-\lambda+1} - \psi_{k}(u_{1}, \ldots, u_{k-\lambda}); (u, \psi_{k}) \in U^{N} \times \Psi_{k}\}
\]

What this definition means is that there are a number of smaller games \( \omega_{\lambda}, \omega_{\lambda+1}, \ldots, \omega_{N} \) --- \( S = N - \lambda + 1 \) in all. These will be called semi-games. The evader must choose his strategy \( u \) from \( U^{N} \) the set of all \( K \)-ary \( N \)-tuples:
\[ u = (u_1, u_2, \ldots, u_N) \in U^N \]

As before. Then, without knowing \( u \), the marksman chooses \( \psi_\lambda \) from the set \( \Psi_\lambda \) which contains his \( M_\lambda \) possible choices, plays the semi-game \( \omega_\lambda \) and obtains the payoff \( h(u_\lambda + u_{\lambda-1} + \ldots + u_1 - \psi_\lambda) \). He then still without knowing \( u \) chooses a \( \psi_{\lambda+1} \) from the set \( \Psi_{\lambda+1} \) which contains his \( M_{\lambda+1} \) possible choices, plays the semi-game \( \omega_{\lambda+1} \), and obtains a payoff \( h(u_{\lambda+1} + u_\lambda + \ldots + u_2 - \psi_{\lambda+1}(u_1)) \). He then continues to \( \omega_{\lambda+3} \) and so on. The process terminates when still without knowing \( u \) he chooses a \( \psi_N \) from the set \( \Psi_N \) of possible choices (of which there are \( M_N \)), plays the semi-game \( \omega_N \) and obtains the payoff \( h(u_N + u_{N-1} + \ldots + u_{N-\lambda+1} - \psi_N(u_1, u_2, \ldots, u_{N-\lambda})) \). The marksman attempts to maximize each individual payoff and thereby maximizes their sum \( J \).

The payoff matrix \( P' \) of the game in semi-normalized form may be partitioned into \( S \) submatrices:

\[
P' = \begin{bmatrix}
P_\lambda & & & \\
- & P_{\lambda+1} & & \\
- & - & - & \ddots \\
- & - & - & \ddots & - \\
- & - & - & \ddots & \ddots \\
- & - & - & \ddots & \ddots & - \\
& & & & & \ddots \\
& & & & & \ddots \\
& & & & & \ddots \\
& & & & & \ddots \\
P_N & & & & & 
\end{bmatrix}
\]

(5-1)
where $P_k$ is the payoff matrix for the semi-game $\omega_k \ k = \lambda, \ldots, N$. $P_k$ contains $K^N$ columns, one for each element of $U^N$, and $M_k = L^{k-\lambda}$ rows, one for each element of $\psi_k$. The number written at the intersection of a particular row and a particular column of $P_k$ is the payoff:

$$h(u_k + u_{k-1} + \ldots + u_{k-\lambda+1} - \psi_k(u_1, u_2, \ldots, u_{k-\lambda}))$$

corresponding to the $u = (u_1, u_2, \ldots, u_N)$ of that column and the $\psi_k$ of that row.

In usage, the evader's strategy $u$ is represented by a column in $P'$. The marksman's strategy $\psi = \{\psi_\lambda, \psi_{\lambda+1}, \ldots, \psi_N\}$ is given by a set of rows in $P'$ consisting of one row from each sub-matrix, that is, the marksman selects one row from the $M_\lambda$ rows of $P_\lambda$, one from the $M_{\lambda+1}$ rows of $P_{\lambda+1}$, and so forth. The payoff $J$ is the total of the entries in the evader's column at each of the rows of the marksman.

The total number of columns of $P'$ is $K^N$. The total number of its rows is:

$$M' = \sum_{k=1}^{N} M_k = \sum_{i=0}^{S-1} L^{k-i}$$

Now, at last, it becomes clear why all this work has been necessary — $M'$ is the sum of the $M_k$ rather than their product (cf. equation (4-4)).

For the game mentioned before in which $K = \lambda = 2$ and $N = 8$, the payoff matrix $P'$ contains 256 columns and approximately $3.43 \times 10^{30}$ rows. This represents a great victory! It is a reduction in the size
of the payoff matrix by a factor of about $10^{30}$! If one had begun writing $P'$ at the creation, he would have completed roughly $2 \times 10^{-12}$ percent by now. Compared to his compatriot writing $P$, he would be nearly finished.

5.2 Example

As an example of the semi-normalized form, consider the case of $K = \lambda = 2$ and $N = 3$. Here $S$, the number of shots, is 2 and $L$, the number of the marksman's choices at each time, is 3.

The decision the evader must make is the choice of a binary 3-tuple $u$ from the set $U^3$ of all binary 3-tuples. There are $2^3 = 8$ possibilities:

$$U^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}.$$

The marksman must make a sequence of $S = 2$ decisions. The first is a choice of $\psi_2$ of which there are $M_2 = L = 3$ possibilities:

$$\psi_2 = \{0, 1, 2\}.$$

The second is a choice $\psi_3(u_1)$ of which there are $L$ possibilities for each possible $u_1$ hence $M_3 = L^2 = 9$ total possibilities:

$$\psi_3 = \{(\psi_3(0), \psi_3(1)) = 00, 01, 02, 10, 11, 12, 20, 21, 22\}.$$

The resulting semi-normalized payoff matrix $P'$ is shown in Figure 5-1. If the evader chooses column $u = (u_1 u_2 u_3) = 010$ and the marksman (who has to make $S = 2$ decisions: he must choose a row from $P_2$ and one from $P_3$) chooses row 2 from $P_2$ and row 12 from $P_3$, he would get payoff 0 from $P_2$ and 1 from $P_3$ for a total payoff:

$$J = 0 + 1 = 1.$$
5.3 The Sub-Normalized Form

Despite the great conceptual and technical advantages of the semi-normalized form, it still has one disadvantage. It is, of course, the fact that $P'$ still has far, far too many rows to be "convenient".

The problem is, that while $M'$ is only the sum rather than the product of the $M_k$, for some values of $k$, $M_k$ is extremely large. The reason is shown by equation (4-3): $M_k$ is equal to $L$ raised to the $k^{-\lambda}$ power and for large values of $k$ the exponent $k^{-\lambda}$ is very large. If only it were $L$ times $k^{-\lambda}$ rather than $L$ to the $k^{-\lambda}$ power . . .

In each semi-game $\omega_k$, the marksman's choice is $\psi_k$. This means that since:

$$\psi_k : U^{k^{-\lambda}} \rightarrow Z$$
the marksman must choose an element from the range $Z$ (which contains a total of $I = \lambda(K-1) + 1$ elements) for each possible $k$-ary $k-\lambda$-tuple $(u_1, u_2, \ldots, u_{k-\lambda})$. It will be shown below how this large single decision may be decomposed into a sequence of smaller decisions.

As before, the simplification is only conceptual — the resulting payoff matrix will simply be easier to write down. The dimensionality of the marksman's decision space is still the same: huge.

The trick lies in defining a generalized hit function:

$$H_{k\nu}(u, \psi_k(v)) = \begin{cases} h(u_k + \ldots + u_{k-\lambda+1} - \psi_k(v)) & \text{if } u_i = v_i \\ 0 & \text{otherwise} \end{cases} \quad i = 1, 2, \ldots, k-\lambda$$

where $v$ is the $K$-ary $k-\lambda$-tuple $v = (v_1, v_2, \ldots, v_{k-\lambda})$. Thus $v \in U^{k-\lambda}$. For convenience, the symbol $U^0$ is defined to mean the set containing the single element 0, and the function $\psi_k(v)$ $v \in U^0$ to be simply the number $\psi_k$.

The sub-normalized form is defined to be the set $\Omega''$:

$$\Omega'' = \bigcup_{k=\lambda}^{N} \bigcup_{v \in U^{k-\lambda}} \omega_{k\nu}$$

(5-4)

where $\omega_{k\nu}$ is the set:

$$\omega_{k\nu} = \{H_{k\nu}(u, \psi_k(v)); (u, \psi_k(v)) \in U^N \times Z\}.$$  \quad (5-5)

What this definition means is that there are a number of small games $\omega_{k\nu} \; v \in U^{k-\lambda} \; k = \lambda, \lambda+1, \ldots, N$ called sub-games. The evader must choose his strategy $u$ from $U^N$ the set of all $K$-ary $N$-tuples, as before.
Then, without knowing \( u \), the marksman chooses \( \psi_{\lambda} \) from the set \( Z \) which contains his \( L \) possible choices and plays the sub-game \( \omega_{\lambda 0} \) for which the payoff is \( H_{\lambda 0}(u, \psi_{\lambda}) \). He then --- still without knowing \( u \) --- plays each of the \( K \) sub-games \( \omega_{\lambda + 1v} \) \( v \in U \) by choosing each \( \psi_{\lambda + 1}(v) \) one at a time from \( Z \) which contains his \( L \) possible choices. The payoff for each of these games is \( H_{\lambda + 1v}(u, \psi_{\lambda + 1}(v)) \) He then continues to the larger set of \( K^2 \) sub-games \( \omega_{\lambda + 2v} \) \( v \in U^2 \) by choosing a \( \psi_{\lambda + 2}(v) \) sequentially for each \( v \in U^2 \) from the \( L \) possible choices in \( Z \). The payoff for each of these games is \( H_{\lambda + 2v}(u, \psi_{\lambda + 2}(v)) \). The process continues. For each sub-game, the marksman tries to maximize the payoff. At the end, the total payoff is the sum of all of the individual payoffs:

\[
J = \sum_{k=\lambda}^{N} \sum_{v \in U^{k-\lambda}} H_{kv}(u, \psi_{k}(v)) .
\]

The payoff matrix \( P^u \) for the sub-normalized form is partitioned horizontally into \( S \) submatrices as before except that each submatrix is itself partitioned into a number of sub, submatrices. Each of these sub, submatrices \( P_{kv} \) has \( L = \lambda (K-1) + 1 \) rows and \( K^N \) columns and is the payoff matrix for the sub-game \( \omega_{kv} \) \( v \in U^{k-\lambda} \) \( k = \lambda, \lambda + 1, ..., N \). \( P_{kv} \) contains \( K^N \) columns, one for each element of \( U^N \) and \( L \) rows, one for each element of \( Z \). The number at the intersection of a particular row and a particular column is the payoff:

\[
H_{kv}(u, \psi_{k}(v))
\]

corresponding to the \( u = (u_1, u_2, ..., u_N) \) of that column and the \( \psi_{k}(v) \) of that row.
In usage, the evader's strategy is represented by a column in $P''$. The marksman's strategy is represented by the combination of a number of rows in $P''$, one row from each of the sub-submatrices $P_{kv}$, that is, the marksman selects one row from the $L \times K^N$ matrix $P_{\lambda 0}$, one row from each of the $L \times K^N$ matrices $P_{\lambda+1 0}$, $P_{\lambda+1 1}$, ..., $P_{\lambda+1 K}$, one row from each of the $L \times K^N$ matrices $P_{\lambda+2 00}$, $P_{\lambda+2 01}$, ..., $P_{\lambda+2 KK}$, and so forth. The payoff $J$ is the total of the entries in the evader's column at each of the rows chosen by the marksman.

The total number of columns in $P''$ is $K^N$. The total number of its rows is:

\[
\begin{bmatrix}
P_{\lambda 0} \\
P_{\lambda+1 0} \\
P_{\lambda+1 1} \\
\vdots \\
P_{\lambda+1 K} \\
P_{\lambda+2 00} \\
P_{\lambda+2 01} \\
\vdots \\
P_{\lambda+2 KK} \\
\vdots \\
P_N 00\ldots 0 \\
P_N KK\ldots K
\end{bmatrix}
\]

(5-6)
\[ M'' = \sum_{k=\lambda}^{N} \sum_{i=0}^{k-\lambda} L = \sum_{k=\lambda}^{N} Lk^{k-\lambda} = L \left( \frac{K^{n-1}}{K-1} \right). \]  

Now it becomes clear why all of this work has been necessary. \( M'' \) is the sum of a number of terms each of which is \( L \) times \( k^{k-\lambda} \) rather than \( L \) to the \( k^{k-\lambda} \) power, (cf. equations (4-4) and (5-2)).

For the game considered before in which \( K = \lambda = 2 \) and \( N = 8 \), the payoff matrix \( P'' \) contains 256 columns and 381 rows. Although not exactly small, this matrix can be easily written down without an extensive expenditure of either pencil lead or time and is certainly small enough to be easily digested by a computer.

5.4 Example

As an example of the sub-normalization technique, consider the case of \( K = \lambda = 2 \), \( N = 4 \). Here, \( S \) the number of shots is 3 and \( L \) the number of marksman's choices at each time is 3.

As before, the decision the evader must make is the choice of a binary 4-tuple \( u \) from the set \( U^4 \) of all binary 4-tuples. There are \( 2^4 = 16 \) possibilities:

\[ U^4 = \{0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111\}. \]

The marksman must make a sequence of \( \left( \frac{K^S-1}{K-1} \right) \) = 7 decisions. He must choose:

\[ \psi_2, \psi_3(0), \psi_3(1), \psi_4(0,0), \psi_4(0,1), \psi_4(1,0), \psi_4(1,1). \]
Each of these decisions must be made from among $l$ choices, the elements of $Z$:

$$Z = \{0, 1, 2\}.$$

The resulting sub-normalized payoff matrix $P''$ is shown in Figure 5-2. Note that since $P''$ is very sparse, only the non-zero elements are shown in the figure. Thus, all blanks should be assumed to be zeros.

Suppose the evader's decision is $u = 0010$, i.e., he decides that his controls will be:

$$u_1 = 0, u_2 = 0, u_3 = 1, u_4 = 0.$$

Suppose also that the marksman chooses rows:

$$\psi_2 = 0 \text{ (the first row of } P_{20}), \quad \psi_3(0) = 1 \text{ (the second row of } P_{30}), \quad \psi_3(1) = 2 \text{ (the third row of } P_{31}), \quad \psi_4(0,0) = 2 \text{ (the third row of } P_{40}), \quad \psi_4(0,1) = 1 \text{ (the second row of } P_{41}), \quad \psi_4(1,0) = 0 \text{ (the first row of } P_{42}), \quad \psi_4(1,1) = 2 \text{ (the third row of } P_{43}).$$

Then the payoff $J$ will be the sum of the entries in the evader's column and the marksman's rows:

$$J = 1 + 1 + 0 + 0 + 0 + 0 + 0 + 0 = 2$$

i.e., there will be two "direct hits".
Figure 5-2: Example of sub-normalized payoff matrix $P_n$. 

\[
\begin{array}{c|c|c|c|c|c|c|c}
 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hline
2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\
\end{array}
\]
5.5 Advantages of the Sub-Normalized Form

In Chapter IV, the disadvantages of the normalized form of aiming and evasion games were pointed out. Although it was pivotal in proving the existence of solutions of these games, the normalized form is far, far too large to be of any practical use.

In this chapter, advantage was taken of the information structure of the Austere Game and the additive nature of its payoff to develop a "not-quite-so-normalized" form which is not as lengthy. Called the "sub-normalized form", it has several advantages. The first is notational — the size of the sub-normalized form for most Austere Games is orders of magnitude smaller than the normalized form so that it is feasible to write it down and analyze it perhaps with the aid of a computer.

The second advantage is conceptual. The sub-normalized form is a technique for compactly displaying all possible outcomes and emphasizing their interrelationships. The game is completely defined by its sub-normalized form and can be analyzed accordingly. By its use, a great deal of the structure of the game is spotlighted. In the simple case in section 5.4, $P''$ has a strong and obvious structure. Note, for example, that each column of $P''$ in Figure 5-2 has exactly three ones in it which reflects the fact that, no matter what strategy (column) is chosen by the evader, there exists a strategy (set of rows) for the marksman by which he can achieve exactly one "hit" with each shot, and there are $S = 3$ shots. In this case, the structure has a
simple explanation. In all cases, however, the notation allows the structure to be readily identified and considered.

A third advantage is, as will be shown in Chapter VI, that some of the known techniques for solving games numerically can, with minor modifications, be applied to games in sub-normalized form. This advantage can hardly be overemphasized since a solution is the object of the analysis.
CHAPTER VI

TECHNIQUES FOR SOLUTION

It is unfortunate but true that general analytical solutions of
Austere Games do not presently exist. To be sure, an exact solution
is presented in Chapter X for the case in which K and λ are both
equal to two. Approximate solutions for wider classes of games also
exist and are given in Chapter IX. Still, the lack of an analytical
solution makes the possibility of a numerical solution more attractive.
In this chapter, two numerical techniques for solving Austere Games
are presented.

The chapter opens in section 6.1 with a discussion of the concept
of a mixed strategy and the expression of the payoff as a bilinear
form. These concepts, it is shown, are directly extendable to games
in sub-normalized form and the details of the extension are set forth.
With this background, the technique of solving games in sub-normalized
form by linear programming is developed in section 6.2 and illustrated
by an example in section 6.3. Since the size of the resulting linear
program grows exponentially with the duration of the game N, a second
technique, the functional equation formulation of dynamic programming,
is presented in section 6.4 and exemplified in section 6.5. This
method will solve long duration games although for large K and λ its
performance is still limited by the "curse of dimensionality".
Conclusions are given in section 6.6.
6.1 Mixed Strategies

The concept of mixed strategies and the expression of the payoff in terms of a bilinear form is directly extendable to games in sub-normalized form. The section will show how this can be accomplished. First, however, it is useful to recall how it is done for the game in normalized form.

A mixed strategy for the evader is a probability density function defined over $U^N$, the set of the evader's open-loop strategies. Since $U^N$ is a finite set containing $K^N$ elements, this is a discrete density. Let the probabilities be:

$$p_\alpha = \begin{cases} 0, 1, \ldots, K^N-1 \end{cases}$$

where $\alpha$ is an index on the elements of $U^N$, i.e., $\alpha = 0$ denotes the 1st element in $U^N$, $\alpha = 1$ represents the 2nd element in $U^N$, while $\alpha = 22$ denotes the 23rd element of $U^N$. It is no coincidence that this index $\alpha$ is the decimal equivalent of the particular element of $U^N$ which it represents, e.g., if $K = 2$, the 23rd element of $U^6$ is:

$$u = 010110$$

which, interpreted as a binary number and converted to decimal, is 22. Another way of saying this is:

$$\Pr\{u \text{ is the } \alpha+1 \text{ element of } U^N\} = \Pr\{(u)_{10} = \alpha\} = p_\alpha$$

$$\alpha = 0, 1, \ldots, K^N-1$$
where the symbol \((u)_{10}\) means "the decimal equivalent of u", i.e.,
\((u)_{10}\) is the result if \(u\) is interpreted as a \(K\)-ary number and converted to base ten. Of course, the \(p_\alpha\) must satisfy:

\[
0 \leq p_\alpha \leq 1 \quad \alpha = 0, 1, \ldots, K^N-1 \\
\sum_\alpha p_\alpha = 1. \tag{6-2}
\]

Similarly, the marksman's mixed strategy is a discrete probability density function over the set of marksman's strategies \(\Psi\). Let it be:

\[
r_\beta \quad \beta = 0, 1, \ldots, M-1
\]

where \(\beta\) is an index on the elements of \(\Psi\) and \(M\), given by equation (4-4), is the total number of elements in \(\Psi\) so that:

\[
Pr\{ \psi \text{ is the } \beta+1 \text{ element of } \Psi \} = r_\beta \quad \beta = 0, 1, \ldots, M-1 \tag{6-3}
\]

and, as usual:

\[
0 \leq r_\beta \leq 1 \quad \beta = 0, 1, \ldots, M-1 \\
\sum_\beta r_\beta = 1. \tag{6-4}
\]

Now if the element in the \(\beta\)th row and \(\alpha\)th column of the payoff matrix \(P\) (the normalized form) is:

\[
a_\beta^\alpha \quad \beta = 0, 1, \ldots, M-1 \\
a_\beta^\alpha \quad \alpha = 0, 1, \ldots, K^N-1
\]
then the expected payoff is the bilinear form:

\[ J = \sum_{i} \sum_{j} a_{ij} r_i p_j. \]  

(6-5)

With this notation, von Neumann's fundamental theorem is easily stated: For any set of \( a_{ij} \) at all, the min max of \( J \) over \( p_j \) and \( r_i \) respectively, subject to constraints (6-2) and (6-4), is equal to the max min over \( r_i \) and \( p_j \) respectively, subject to the same constraints.

This idea of a mixed strategy and its notation is easily extended to the sub-normalized form of the game. The definition of the mixed strategy density function \( p_{\alpha} \), \( \alpha = 0, 1, \ldots, k^N - 1 \) for the evader remains precisely the same. However the corresponding one for the marksman must be modified.

The marksman's mixed strategy density function for the sub-normalized form is:

\[ \gamma = 0, 1, \ldots, L-1 \]

\[ r_{k^\beta \gamma} \]

\[ \beta = 0, 1, \ldots, k^{k-\lambda} - 1 \]

\[ k = \lambda, \lambda+1, \ldots, N \]

defined by:

\[ \Pr[\psi_k(v) = \gamma \text{ where } v \text{ is the } \beta+1 \text{ element of } U^{k-\lambda}] = \Pr[\psi_k(v) = \beta \text{ where } (v)_1 = \beta] = r_{k^\beta \gamma}. \]

Notice that, as before, \( \beta \) the index over the elements of \( U^{k-\lambda} \) is also, by design, equal to their decimal equivalents.
The axioms of probability are:

\[ 0 \leq r_{k\beta \gamma} \leq 1 \]
\[ \gamma = 0, 1, \ldots, L-1 \]
\[ \beta = 0, 1, \ldots, k^{k-\lambda}-1 \]
\[ \sum_{\gamma} r_{k\beta \gamma} = 1 \]
\[ k = \lambda, \lambda+1, \ldots, N \]  \hspace{1cm} (6-6)

and, denoting the element in row \( \gamma \), column \( \alpha \) of \( P'_{k\beta} \) in the sub-normalized payoff matrix \( P'' \) by:

\[ a_{k\beta \gamma \alpha} \]
\[ \alpha = 0, 1, \ldots, K^{N-1} \]
\[ \gamma = 0, 1, \ldots, L-1 \]
\[ \beta = 0, 1, \ldots, k^{k-\lambda}-1 \]
\[ k = \lambda, \lambda+1, \ldots, N \]

the payoff becomes:

\[ J = \sum_{k} \sum_{\beta} \sum_{\gamma} \sum_{\alpha} a_{k\beta \gamma \alpha} r_{k\beta \gamma} P_{\alpha} \]  \hspace{1cm} (6-7)

which is still a bilinear form to which von Neumann's fundamental theorem still applies. The Value of the game is:

\[ V = \min_{P_{\alpha}} \max_{r_{k\beta \gamma}} J \]  \hspace{1cm} (6-8a)
\[ = \max_{r_{k\beta \gamma}} \min_{P_{\alpha}} J \]  \hspace{1cm} (6-8b)

6.2 Solution by Linear Programming

It is well known that games in normalized form can be solved by linear programming. Indeed, this is one of the most efficient known methods for solving general games. The book by Hadley [21] contains
a readable exposition of the technique. Since this material is readily available, the discussion here will be restricted to a linear programming formulation of the Austere Game in sub-normalized form.

Although, computationally, games can usually be solved all at once, it is best from a pedagogical point of view to split the discussion into two parts: the solution for the evader's strategy, and the solution for the marksman's.

With the help of equations (6-6), equations (6-7) and (6-8a) can be combined to read:

$$V = \min_{\alpha} \sum_{k} \sum_{\beta} \max_{\gamma} \sum_{\alpha} a_{k\beta\gamma\alpha} p_{\alpha}.$$  

Now, defining:

$$q_{k\beta} = \max_{\gamma} \sum_{\alpha} a_{k\beta\gamma\alpha} p_{\alpha} \quad (6-9)$$

and keeping in mind equations (6-2), yields:

**Primal Linear Programming Problem:**

$$V = \min_{\alpha} q_{k\beta} \sum_{k} \sum_{\beta} q_{k\beta} \quad (6-10a)$$

subject to:

$$\sum_{\alpha} a_{k\beta\gamma\alpha} p_{\alpha} - q_{k\beta} \leq 0 \quad \gamma = 0, \ldots, L-1$$

$$\beta = 0, \ldots, k^{k-\lambda-1} \quad (6-10b)$$

$$k = \lambda, \ldots, N \quad (6-10c)$$

$$\sum_{\alpha} p_{\alpha} = 1$$
and:

\[ p_\alpha \geq 0 \quad \alpha = 0, 1, \ldots, K^{N-1}. \] (6-10d)

The reason that this has been called the Primal Linear Programming Problem will become evident momentarily. Notice that the solution of this problem furnishes the Value V and the evader's optimal strategy \( p_\alpha \quad \alpha = 0, 1, \ldots, K^{N-1} \) -- however it gives no information at all about the marksman's optimal strategies.

The variables \( q_{k\beta} \) admit an interesting interpretation. Comparing (6-10a) and (3-13), it can be seen that:

\[ \sum_{\beta} q_{k\beta} = \text{Pr}\{\text{"direct hit" at time } k\} \] (6-11)

and remembering what the index \( \beta \) is, therefore:

\[ q_{k\beta} = \text{Pr}\{\text{"direct hit" at time } k \text{ and } (u_1, u_2, \ldots, u_{k-\lambda})_{10} = \beta}\}. \] (6-12)

The size of this LP Problem is quickly calculated. There are \( K^N \) of the \( p_\alpha \) variables and one \( q_{k\beta} \) for each possible combination of \( k \) and \( \beta \) of which there are:

\[ \frac{K^S}{(K-1)}. \]

Similarly, there are \( L = \lambda(K-1) + 1 \) equations in (6-9b) for each possible combination of \( k \) and \( \beta \). Thus, the size of the Primal Problem is:
**65**

\( p_\alpha \) variables (non-negative): \( \binom{N}{K} \)

\( q_{k\beta} \) variables (unconstrained): \( \frac{\binom{S}{K-1}}{K-1} \)

inequality constraints: \( L \binom{S}{K-1} \)

equality constraints: 1

(6-13)

This is usually a large problem for present day linear programming codes. However, many small sized problems can be easily handled.

For example, the game discussed in Chapters IV and V for which \( K = \lambda = 2, N = 8, S = 7, L = 3 \), has:

\( p_\alpha \) variables (non-negative): 256

\( q_{k\beta} \) variables (unconstrained): 127

inequality constraints: 381

equality constraints: 1

But, one might ask, how can the marksman's optimal mixed strategy be found? The answer will be shown now to involve solving an LP problem dual to the one above.

With the help of equations (6-2), equations (6-7) and (6-8b) can be combined to yield:

\[ V = \max_{r_k} \min_{\beta, \gamma} \sum_k \sum_{\beta} \sum_{\gamma} a_{k\beta\gamma} r_{k\beta\gamma} \cdot \]

Now, defining:

\[ w = \min_{\alpha} \sum_k \sum_{\beta} \sum_{\gamma} a_{k\beta\gamma} r_{k\beta\gamma} \quad (6-14) \]

and keeping in mind equations (6-6), yields:
Dual Linear Programming Problem:

\[ V = \max_{r_k, \beta, \gamma} w \]

subject to:

\[ \sum_k \sum_{\beta} \sum_{\gamma} a_{k,\beta,\gamma} r_{k,\beta,\gamma} - w \geq 0 \quad \alpha = 0, 1, \ldots, K^{N-1} \]  

\[ \sum_{\gamma} r_{k,\beta,\gamma} = 1 \quad \beta = 0, 1, \ldots, K^{k-\lambda-1} \]

\[ k = \lambda, \lambda+1, \ldots, N \]

and:

\[ r_{k,\beta,\gamma} \geq 0 \quad \gamma = 0, 1, \ldots, L-1 \]

\[ \beta = 0, 1, \ldots, K^{k-\lambda-1} \]

\[ k = \lambda, \lambda+1, \ldots, N \]

The size of this LP Problem is quickly calculated:

- \( r_{k,\beta,\gamma} \) variables (non-negative): \( L \binom{K^{S-1}}{K-1} \)
- \( w \) variables (unconstrained): 1
- Inequality constraints: \( K \)
- Equality constraints: \( \binom{K^{S-1}}{K-1} \).

Again, this is a large but in many cases conceivable linear programming problem. In the case that has been discussed where \( K = \lambda = 2 \) and \( N = 8 \), the results are:

- \( r_{k,\beta,\gamma} \) variables (non-negative): 381
- \( w \) variables (unconstrained): 1
- Inequality constraints: 256
- Equality constraints: 127.
Notice that these two problems are duals in the linear programming sense: The non-negative Dual variables $r_{k\beta Y}$ are dual to the Primal inequality constraints; the unconstrained Dual variable $w$ is dual to the Primal equality constraint; the non-negative Primal variables $p_{a}$ are dual to the Dual inequality constraints; and the unconstrained Primal variables $q_{k\beta}$ are dual to the Dual equality constraints. Also, the Primal is a minimization problem while the Dual is a maximization one; and the sense of the inequalities in the Dual are opposite to those in the Primal. In fact, in every sense, the two are indeed the duals of each other.

Any LP code used to solve these problems should take advantage of the fact that a feasible solution, many feasible solutions in fact, exist and are easily found for both Primal and Dual. The proof is by construction: To find a feasible solution to the Primal simply choose any set of $p_{a}$ which satisfy equality constraint (6-10c) and non-negativity constraints (6-10d). These in combination with the set of $q_{k\beta}$ found by use of equation (6-9) constitute the desired feasible solution of the Primal. Similarly, any set of $r_{k\beta Y}$ which satisfies equality constraints (6-15c) and non-negativity constraints (6-15d) together with $w$ found by equation (6-14) constitute a feasible solution of the Dual.

From this result, there follows another proof of the existence of a saddle-point: Since there exist a feasible solution to the Primal and a feasible solution to the Dual, it follows from linear
programming duality theory that an optimal solution for each exists and that they are equal. Now, from the derivation of the Primal it can be seen that its optimal solution is (cf. equation (6-8a)):

$$\min_{P_\alpha} \max_{r_{k\beta\gamma}} J$$

while the optimal solution to the Dual is (cf. equation (6-8b)):

$$\max_{r_{k\beta\gamma}} \min_{P_\alpha} J$$

so that therefore:

$$\min_{P_\alpha} \max_{r_{k\beta\gamma}} J = \max_{r_{k\beta\gamma}} \min_{P_\alpha} J$$

and therefore by equation (3-6) a saddle-point exists which was to be shown.

Linear programming theory also says that the optimal mixed strategy need not be unique for either player. However, if they are not, the set of all optimal mixed strategies for each player form a closed, convex set. Also, no more than:

$$\min \left[ K^N + \frac{K^S - 1}{K - 1}, \ L\frac{K^S - 1}{K - 1} + 1 \right]$$

of the variables: $p_\alpha$ and $q_{k\beta}$ or of the variables: $r_{k\beta\gamma}$ and $w$ need be different from zero.
6.3 Example

Figures 6-1 and 6-2 show the LP formulation of the game which was used as an example in section 5.4 of Chapter V for which \( K = \lambda = 2 \), \( N = 4 \). Matrix notation has been used in the figures where: \( p \) is a column vector of the \( p_\alpha \), \( \alpha = 0, 1, \ldots, K^{N-1} \); \( r \) is a column vector of the \( r_{k\beta} \), \( \gamma = 0, 1, \ldots, L-1, \beta = 0, \ldots, K^{k-\lambda}-1, k = \lambda, \ldots, N \); and \( q \) is a column vector of the \( q_{k\beta} \), \( \beta = 0, \ldots, K^{k-\lambda}-1, k = \lambda, \ldots, N \).

In this case, \( p \) is a 16-vector, \( r \) is a 21 vector, and \( q \) is a 7-vector. The matrices in Figures 6-1 and 6-2 contain only zeros and ones, however since they are fairly sparse, only the ones have been shown. All blank spots in the matrices contain zeros.

This problem was actually solved on a Digital Equipment Corporation PDP-9 digital computer using a linear programming code written for the PDP-9 by Professor Kenneth J. Breeding of Ohio State University. The solution is shown in Figure 6-3. The hit probability per shot \( V/S \) of about .352 compares favorably with the .382 obtained by Isaacs for Isaacs' Game described in Chapter III.

6.4 Solution by Dynamic Programming

The linear programming technique, useful as it is, has several basic disadvantages. For one, it seems only useful for small sized games since even the largest digital computers cannot solve linear programs which have more than a few thousand rows. Secondly, the
Minimize: \( \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} q \)

subject to:

\[
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\end{bmatrix}
\begin{bmatrix}
p \\
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 \\
1 & 1 \\
1 \\
\end{bmatrix} \\
\begin{bmatrix}
p \geq \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\end{bmatrix} \\
\begin{bmatrix}
p = 1 \\
\begin{bmatrix}
p \geq 0 \\
\end{bmatrix} \end{bmatrix}
\]

Figure 6-1 Example of Primal Linear Programming Problem.
Maximize: \[ w \]

subject to:

\[
\begin{bmatrix}
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
1 & 1 & & & & & \\
\end{bmatrix}
\begin{bmatrix}
r_1 \\
r_2 \\
r_3 \\
r_4 \\
r_5 \\
r_6 \\
r_7 \\
r_8 \\
r_9 \\
r_{10} \\
r_{11} \\
r_{12} \\
\end{bmatrix}
\geq
\begin{bmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
\end{bmatrix}
\]

and:

\[
\begin{bmatrix}
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & & & & & & \\
\end{bmatrix}
\begin{bmatrix}
r_1 \\
r_2 \\
r_3 \\
r_4 \\
r_5 \\
r_6 \\
r_7 \\
r_8 \\
r_9 \\
r_{10} \\
r_{11} \\
r_{12} \\
\end{bmatrix}
= 
\begin{bmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
r_1 \\
r_2 \\
r_3 \\
r_4 \\
r_5 \\
r_6 \\
r_7 \\
r_8 \\
r_9 \\
r_{10} \\
r_{11} \\
r_{12} \\
\end{bmatrix}
\geq
0
\]

Figure 6-2  Example of Dual Linear Programming Problem
\[
\begin{bmatrix}
 p_0 & p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & p_{11} & p_{12} & p_{13} & p_{14} & p_{15} \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
 Pr\{u=0000\} & Pr\{u=0001\} & Pr\{u=0010\} & Pr\{u=0011\} & Pr\{u=0100\} & Pr\{u=0101\} & Pr\{u=0110\} & Pr\{u=0111\} & Pr\{u=1000\} & Pr\{u=1001\} & Pr\{u=1010\} & Pr\{u=1011\} & Pr\{u=1100\} & Pr\{u=1101\} & Pr\{u=1110\} & Pr\{u=1111\} \\
1/9 & 1/9 & 0 & 1/9 & 1/9 & 0 & 1/9 & 1/9 & 0 & 0 & 0 & 1/9 & 1/18 & 1/18 & 1/9 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
 r_{200} & r_{201} & r_{202} & r_{203} & r_{204} & r_{205} & r_{206} & r_{207} & r_{208} & r_{209} & r_{210} & r_{211} & r_{212} & r_{213} & r_{214} \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
 Pr\{\psi_2=0\} & Pr\{\psi_2=1\} & Pr\{\psi_2=2\} & Pr\{\psi_3(0)=0\} & Pr\{\psi_3(0)=1\} & Pr\{\psi_3(0)=2\} & Pr\{\psi_3(1)=0\} & Pr\{\psi_3(1)=1\} & Pr\{\psi_3(1)=2\} & Pr\{\psi_4(00)=0\} & Pr\{\psi_4(00)=1\} & Pr\{\psi_4(00)=2\} & Pr\{\psi_4(01)=0\} & Pr\{\psi_4(01)=1\} & Pr\{\psi_4(01)=2\} & Pr\{\psi_4(10)=0\} & Pr\{\psi_4(10)=1\} & Pr\{\psi_4(10)=2\} & Pr\{\psi_4(11)=0\} & Pr\{\psi_4(11)=1\} & Pr\{\psi_4(11)=2\} \\
2/9 & 5/9 & 2/9 & 1/2 & 1/2 & 0 & 0 & 1/2 & 1/2 & 1/3 & 1/3 & 1/3 & 0 & 1/2 & 1/2 & 1/2 & 1/2 & 0 & 1/3 & 1/3 & 1/3 \\
\end{bmatrix}
\]

\[V = \text{Value} = \frac{1}{18}\]

\[V/S = \text{Average hit probability per shot} = \frac{19}{54} = .352\]

Figure 6-3 Optimal solution of linear programming example.
linear programming algorithm is very general—it will work for any arbitrary sub-normalized payoff matrix. However, the examples have shown (sections 5.4 and 6.3) that the payoff matrices for these games have a great deal of structure: a) they are very sparse, b) 1 is the only non-zero element, c) the ones are arranged in distinctive patterns. Thus, one would expect that a general algorithm that does not take advantage of any specialized structure of the payoff matrix would be inefficient compared to one which does. In this section, a dynamic programming technique [22] is developed which does indeed take advantage of some of the structure of these games.

Equation (6-10a) shall be the point of departure:

$$V = \min_{p_\alpha} \sum_k \sum_\beta q_{k\beta}.$$ 

It must be remembered that the minimization is over all $p_\alpha$, $\alpha = 0, 1, \ldots, K^N-1$ which satisfy the axioms of probability (6-10c) and (6-10d). It must also be remembered what the meaning of $q_{k\beta}$ is. Specifically, from (6-11):

$$\sum_\beta q_{k\beta} = \text{Pr("direct hit" at time } k).$$

Now the structure of the problem dictates that this probability is not a function of all the $p_\alpha$ but is only a function of the probabilities of the first $k$ controls:

$$s_{k\xi} = \text{Pr}(u_1, u_2, \ldots, u_k) = \xi+1 \text{ element of } U^k$$

$$= \text{Pr}(u_1, u_2, \ldots, u_k)_0 = \xi \text{ } \xi = 0, 1, \ldots, K^k-1.$$
Obviously, no controls chosen at times later than k have any effect on the hit probability at time k. The probabilities \( s_{k\xi} \) are related to \( p_\alpha \) by:

\[
\begin{align*}
    s_{k\xi} &= \sum_{i=0}^{N-k-1} P(\xi_k N-k+1) \\
&= \sum_{i=0}^{K-1} s_{k+i}(\xi_{k+1})^i
\end{align*}
\]  

and are related among themselves by:

\[
\begin{align*}
    s_{k\xi} &= \sum_{i=0}^{K-1} s_{k+i}(\xi_{k+1})^i
\end{align*}
\]  

Since the \( s_{k\xi} \) are probabilities, they must satisfy the axioms of probability:

\[
\begin{align*}
    0 &\leq s_{k\xi} \leq 1 & \xi = 0, 1, \ldots, K^{k-1} \\
    \sum_{\xi} s_{k\xi} &= 1 & k = \lambda, \lambda+1, \ldots, N.
\end{align*}
\]  

For notational purposes, it will be convenient to define the vector:

\[
    s_k = [s_k 0; s_{k+1}; s_{k+2}; \ldots; s_k K^{k-1}].
\]

Now suppose the function \( F_m(s_{m-1}) \) is defined:

\[
    F_m(s_{m-1}) = \min_{p_\alpha} \sum_{k=m}^{N} \sum_{\beta} q_{k\beta}
\]

whose domain is all \( s_{m-1} \) satisfying equations (6-19) and where the minimization is over all \( p_\alpha \) satisfying (6-10c), (6-10d), and (cf. equation (6-17)): 
\[ \sum_{i=0}^{K^{N-m+1}-1} P(\xi_{k^{N-m+1}+i}) = s_{m-1} \xi \quad \xi = 0, 1, \ldots, K^{m-1}-1. \]

Then, keeping in mind equations (6-18) and (6-19), the functional equation:

\[ F_m(s_{m-1}) = \min_{s_m} \left[ \sum_{\beta} q_{m\beta} + F_{m+1}(s_m) \right] \]

subject to:

\[ \sum_{i=0}^{K-1} s_m(\xi_{k+i}) = s_{m-1} \xi \quad \xi = 0, 1, \ldots, K^{m-1}-1 \]

\[ s_{m, i} \geq 0 \quad i = 0, 1, \ldots, K^{m-1} \]

with:

\[ F_{N+1}(s_N) = 0 \]

is obtained and the Value can be written:

\[ V = \min_{s_{\lambda-1}} F_\lambda(s_{\lambda-1}). \]

The dynamic programming formulation is not actually complete, however, because the relationship of \( q_{m\beta} \) to \( s_m \) has not been explained. From equation (6-12):

\[ q_{m\beta} = \Pr(" \text{direct hit" at time } k \text{ and } u_1, u_2, \ldots, u_{k-\lambda})_{10} = \beta} \]

or:
\[ q_{m\beta} = \max_{\gamma} \Pr\{u_{k-\lambda+1} + u_{k-\lambda+2} + \ldots + u_k = \gamma \text{ and } (u_1 u_2 \ldots u_{k-\lambda})_{10} = \beta\} \]

or:

\[ q_{m\beta} = \max_{\gamma} \sum_{\xi} b_{k\beta\gamma\xi} s_{k\xi} \]  \hspace{1cm} (6-21)

where:

\[ b_{k\beta\gamma\xi} = \begin{cases} 
1 & \text{if } (u_1 u_2 \ldots u_{k-\lambda})_{10} = \beta \\
0 & \text{otherwise}.
\end{cases} \]

Thus is obtained:

*Dynamic Programming Problem I:*

\[ F_m(s_{m-1}) = \min_{s_m, q_{m\beta}} \left[ \sum_{\beta} q_{m\beta} + F_m(s_m) \right] \]  \hspace{1cm} (6-22a)

subject to:

\[ \sum_{\xi} b_{m\beta\gamma\xi} s_{m\xi} - q_{m\beta} \leq 0 \quad \gamma = 0, 1, \ldots, L-1 \]  \hspace{1cm} (6-22b)

\[ \sum_{i=0}^{K-1} s_m (\xi K+i) = s_{m-1} \xi \]  \hspace{1cm} (6-22c)

and:

\[ s_{m\xi} \geq 0 \quad \xi = 0, 1, \ldots, k^{m-1}-1 \]  \hspace{1cm} (6-22d)

with:

\[ F_{N+1}(s_N) = 0. \]  \hspace{1cm} (6-22e)
Here, the restriction in equation (6-19) that the sum of the probabilities must be one has been deferred until the calculation of the Value when:

\[ V = \min_{s_{\lambda-1}} F(s_{\lambda-1}) \]

subject to:

\[ \sum_{\xi} s_{\lambda-1} \xi = 1 \quad s_{\lambda-1} \xi \geq 0 \quad \xi = 0, 1, \ldots, k^{\lambda-1}-1. \]

Solving Dynamic Programming Problem I is equivalent to solving the game sequentially from the end, i.e., the game is first solved at \( k = N \), then at \( k = N - 1 \), etc. Another way of stating this is that solving it is equivalent to solving the Primal Linear Programming Problem sequentially: first minimizing \( \sum_{\beta} q_{N\beta} \) subject to the last \( k^{N-1} \) constraints in (6-10b) plus additional constraints (equation (6-22c)) to allow for the neglected ones in (6-10b); then minimizing the sum of the resulting objective function and \( \sum_{\beta} q_{N-1\beta} \) subject to the next-to-last \( k^{N-\lambda-1} \) constraints in (6-10b) plus additional ones (equation (6-22c)) to allow for the neglected constraints in (6-10b); then the process continues until the whole objective function (6-10a) of the Primal LP Problem has been minimized.

Bellman [22] has pointed out that, computationally speaking, the more constraints of the type (6-22b), (6-22c) and (6-22d) the better, for they simply reduce the volume of space which must be exhaustively searched to find the minimum in (6-22a).
Nevertheless, Dynamic Programming Problem I is of far too high dimensionality to be practical except in small-sized problems. Fortunately, the following result mitigates this unpleasant circumstance.

The function $F_k(s_{k-1})$ is linearly separable:

$$F_k(s_{k-1}) = \sum_{\beta} G_k(s_{k-1}, \eta, s_{k-1} n+1, \ldots, s_{k-1} (n+K-1))$$

(6-23)

where $\eta = \beta K^{\lambda-1}$ and $G_k$ is the solution of:

\text{Dynamic Programming Problem II:}

\begin{align*}
G_m(t_{m-1}, t_{m-1} 1, \ldots, t_{m-1} (K^{\lambda-1}-1)) &= \min_{t_m 0, t_m 1, \ldots, t_m (K^{\lambda-1}), q_m}
\left[ q_m + \sum_{i=0}^{K-1} G_{m+1}(t_m (iK^{\lambda-1}), t_m (iK^{\lambda-1}+1), \ldots, t_m (iK^{\lambda-1}+K^{\lambda-1}-1)) \right]
\end{align*}

subject to:

\begin{align*}
\sum_{j=0}^{K^{\lambda-1}} b_{m0yj} t_{mj} - q_m &\leq 0 & \gamma = 0, 1, \ldots, L-1 \\
\sum_{i=0}^{K-1} t_m (hK+i) &= t_{m-1} h & h = 0, 1, \ldots, K^{\lambda-1}-1 \\
t_{mj} &\geq 0 & j = 0, 1, \ldots, K^{\lambda-1}
\end{align*}

with:

$$G_{N+1}(\ldots) \equiv 0.$$ 

The proof of this result is rather complicated and is developed in Appendix A.
If the set of functions $G_m$ are known, then the Value of the game can be found by:

$$V = \min_{t_{\lambda-1}^{\lambda-1}} G_\lambda(t_{\lambda-1}^{\lambda-1}, t_{\lambda-1}^{\lambda-1}, \ldots, t_{\lambda-1}^{\lambda-1})$$

subject to:

$$\sum_{i=0}^{K-1} t_{\lambda-1}^i = 1$$

$$t_{\lambda-1}^h \geq 0 \quad h = 0, 1, \ldots, K^{\lambda-1}-1.$$

As can be seen, the domain of the function $G_m$ is $K^{\lambda-1}$ dimensional for all $m$. This is usually high and it does not take too large a value of $K$ or $\lambda$ for the "curse of dimensionality" [22] to limit the effectiveness of the method. However, for small values of $K$ and $\lambda$, the Dynamic Programming approach looks practical, especially for games of long duration since the dimension of the domain of $G_m$ is not a function of $N$.

6.5 Example

The Dynamic Programming Formulation will now be written out in detail for the case $K = \lambda = 2$, $N = 4$, the case which has been an example several times previously. For this case, the Dynamic Programming Problem II form is:

$$G_m(x,y) = \min_{t_0, t_1, t_2, t_3} [q + G_{m+1}(t_0, t_1) + G_{m+1}(t_2, t_3)]$$

subject to:
The functions $G_m$ are found sequentially, one at a time, starting with $G_4$, then $G_3$, then $G_2$. When $G_2$ has been found, the Value of the game may be calculated as:

$$V = \min_{0 \leq x \leq 1} G_2(x, 1-x).$$

As can be seen, this simple two dimensional functional equation should be easy to solve with the aid of a digital computer. The Dynamic Programming Formulation is especially useful for games of long duration since the dimensionality of the domain of $G_m$ is independent of $N$.

6.6 Summary and Conclusions

In this chapter two numerical techniques for solving Austere Games were derived. The first, linear programming, is well known as a solver of games in normalized form. It is however applicable to games in sub-normalized form as well and therefore is practical for some Austere Games. The technique involves solving a linear programming problem to find the evader's optimal strategy and the Value
and then solving another linear programming problem, dual to the first, to find the marksman's optimal strategy and the Value (the latter is thus obtained twice). The method works quite well since there exist efficient algorithms for solving linear programs. However, the size of the linear program grows exponentially with the duration of the game N so that the technique is practical only for Austere Games of relatively short duration.

Because of this limitation, the technique of solving Austere Games by Dynamic Programming was presented. This effectively changed the exponential growth of the problem size (and hence the computational difficulty) with N to a linear one. This is because each iterative solution of the dynamic programming functional equation is independent of N but a total of \( S = N - \lambda + 1 \) iterations must be performed. Thus the dynamic programming technique seems much more attractive than linear programming for solving Austere Games of long duration. However, the domain of the Value function in the functional equation grows exponentially with \( \lambda \) so that in games with large \( \lambda \) even dynamic programming falls victim to the "curse of dimensionality".

The linear programming formulation has other benefits besides that of furnishing an efficient computational procedure. For it was by a careful inductive study of the optimal solution to the linear programming example in section 6.3 that the exact analytical solution presented in Chapter X for a whole class of Austere Games was obtained. At this writing the author knows of no other way this
solution could have been obtained. Furthermore, the same technique
should be applicable to other classes of games: one or more
particular examples of the class are solved by linear programming
and then the whole class solved by induction with the aid of the
intuition thus gained. Thus the linear programming formulation is
truly a useful one.
CHAPTER VII

GRAPH THEORETIC CONSIDERATIONS

The two numerical techniques in the last chapter are good ones. However, they take advantage of very little of the special structure of Austere Games. Linear programming for example will solve any game in normalized or sub-normalized form. This includes Austere Games which have a very unique structure typified by Figure 5-2. On the other hand, it might be expected that methods which take advantage of this special structure should be able to solve Austere Games more efficiently or solve wider classes of them. In this chapter some of the groundwork necessary for a further exploitation of the structure of these games will be laid.

Graph theory seems a logical vehicle for the abstraction and study of such structure. Therefore section 7.1 is devoted to a description of the graphs engendered by Austere Games. Section 7.2 delves more deeply into the obvious symmetry of the graphs and therein is contained a general and useful symmetry theorem. In the sequel, it will be shown that the number of paths through these networks is the important factor and section 7.3 introduces a function, the path function, which gives this number. Since this function is so important, it is analyzed there in detail. The chapter ends with summary and conclusions in section 7.4.
7.1 Formulation as a Graph

The Auster Game can be represented in the form of a graph. Let each possible value of the evader's position $x_k$ for each possible time $k = 0, 1, 2, \ldots, N$ be a "node" of the graph. Let a "branch" connect each pair of nodes which represent a possible transition from one time point to the next. That is, if two nodes represent adjacent points in time $k$ and $k-1$ and represent values of the evader's state $x_{k-1}$ and $x_k$ which satisfy:

$$x_k = x_{k-1} + u_k; \quad u_k \in U$$

then they are connected by a "branch".

It is customary to show such graphs in pictorial form. In two dimensions, one axis, say the horizontal one, represents time $k$ with the integer values of $k$ equally spaced and sequentially ordered say from left to right. At each value of time $k$, i.e., at each horizontal position, there is a vertical row of points or "nodes" one for each possible value of $x_k$. As was discussed in Chapter III, $x_k$ can only take on non-negative integer values up to $k(K-1)$ so that there are $k(K-1) + 1$ nodes arranged vertically at each time point $k$. In such a picture each branch is shown as a straight line between the two nodes which it connects. Figure 7-1 shows several graphs.

Notice that:

1. The nodes are arranged in columns, one column for each value of time $k$. This is emphasized in Figure 7-1 by the inclusion of a
Figure 7-1 Examples of graphs.
time axis below each graph indicating the value of \( k \) corresponding to each column.

2. The upper-most node in each vertical column represents the evader state zero, i.e., the upper-most node in column \( k \) represents \( x_k = 0 \). The nodes below it are ordered sequentially with the larger values of \( x_k \) lower in the column. To emphasize this, each node in Figure 7-1 is labeled with the value of \( x_k \) which it represents.

3. Time flows from left to right so that the evader's state "flows" from left to right. Thus the graph may be thought of as "oriented" or "directed" in the sense that only nodes to the right of a given node and connected to it by branches represent legitimate transitions (in the forward time sense). This point is so obvious and the pictorial representation of the graphs so systematic that no effort has been made in the nomenclature to denote it (there are no little arrowheads on the branches for example). It should be kept in mind, however.

4. Each node has \( K \) branches emanating from it (in the forward time sense: to the right) corresponding to the evader's \( K \) choices at that time. These branches are ordered sequentially so that the upper-most one represents \( u_k = 0 \), the one below that \( u_k = 1 \), and so forth. Thus, while each node represents a possible evader state, each branch represents a possible control.

5. There are \( k(K-1) + 1 \) nodes in the \( k \)th column and thus each column
contains K-1 more nodes than its predecessor.

6. A game whose length is N is represented by a graph containing N+1 columns of nodes.

The purpose of graphs is to represent the relationships between nodes. Strictly speaking these are indicated by the branches alone. Thus, the physical relationship of the nodes in the pictures, e.g., drawing them in neat columns, has absolutely no significance. The graphs could be redrawn with the nodes scattered randomly and, so long as all the branches are drawn in, the graph would be essentially the same, in fact, it would be identical. The neat ordering of the nodes into columns in the figure and the emphasis on their ordering in the comments above have been done only for the purpose of standardization and to avoid any clumsy labeling schemes for the nodes. This practice will be continued in the pages which follow but it should be remembered that it is the branches connecting the nodes which contain the information.

The left-most node in the graph, usually representing $x_0 = 0$, shall be called the "origin" of the graph while the right-most column, normally containing all the nodes representing $x_N$, shall be called its "face". Both of these are illustrated on the graphs in Figure 7-1. The graph of a game whose length is N shall be called an N-graph. Often it is desirable to depict only a portion of a graph --- either a portion during a time interval smaller than 0 to N or showing a subset of the possible evader's states. Of particular
interest will be the $\lambda$-graph whose origin is $x_{k-\lambda}$ the last state known to the marksman and whose face is the set of all possible $z_k = x_k - x_{k-\lambda}$.

In other words, the face contains all the possible points at which the evader might be, given his position at $x_{k-\lambda}$ and thus is the set of all possible aiming points $z_k$.

A "path" is an ordered set of branches in a graph connecting a series of nodes representing successive points in time. Since each node represents a possible state of the evader, the series of successive nodes in a path represent successive states of the evader so that each path in the graph represents a possible trajectory for the evader. The branches in a path represent the evader's controls which generate that trajectory. A path of "length" $\ell$ contains $\ell$ branches and $\ell+1$ successive nodes. Since each of the $\ell$ branches in a path represent an evader's control, there are $K^\ell$ possible paths of length $\ell$. They represent a $K$-ary $\ell$-tuple $(u_1, u_2, \ldots, u_\ell)$.

7.2 Symmetry

Perhaps the most striking characteristic of the graphs in Figure 7-1 is their strong symmetry about the horizontal centerline. Indeed, this symmetry is shared by all Austere Games, a fact which will now be demonstrated.

Let $\overline{u_k}$ be the K-1's complement of $u_k$:

$$\overline{u_k} = (K-1) - u_k$$
and let the N-tuple \( \bar{u} = (\bar{u}_1, \bar{u}_2, \ldots, \bar{u}_N) \) be the K-1's complement of the N-tuple \( u \): 
\[
\bar{u} = (\bar{u}_1, \bar{u}_2, \ldots, \bar{u}_N) = (K-1-u_1, \ldots, K-1-u_N).
\]

Notice that if \( u \) is an element of \( U^N \), then so also is \( \bar{u} \):
\[
u_k \in U \quad \bar{u}_k \in U \quad \bar{u} \in U^N.
\]

Now, also let the function \( \bar{\psi}_k \), the L-1's complement of \( \psi_k \), be related to \( \psi_k \) by:
\[
\bar{\psi}_k(u_1, u_2, \ldots, u_{k-\lambda}) = L-1 - \psi_k(u_1, u_2, \ldots, u_{k-\lambda})
\]
\[
= L-1 - \psi_k(K-1-u_1, \ldots, K-1-u_{k-\lambda})
\]
and define \( \bar{\psi} \):
\[
\bar{\psi} = \{ \bar{\psi}_1, \bar{\psi}_{1+1}, \ldots, \bar{\psi}_N \}.
\]

Notice that, as the domain and range of \( \psi_k \) are \( U^{k-\lambda} \) and \( Z \) respectively, so also they are for \( \bar{\psi}_k \):
\[
\bar{\psi}_k : U^{k-\lambda} \rightarrow Z.
\]

The basic symmetry result can now be stated. For the Austere Game:
\[
J(u, \psi) = J(\bar{u}, \bar{\psi}). \quad (7-1)
\]

The proof is obtained by combining equations (3-20) and (3-17) to yield:
\[ J(u, \psi) = E \left[ \sum_{k=1}^{N} h(u_k^+u_{k-1}^+\ldots+u_{k-\lambda+1}^+\psi_k (u_1, u_2, \ldots, u_{k-\lambda})) \right] \]
\[ = E \left[ \sum_{k=1}^{N} h(\lambda-k-1)u_1^+u_2^-\ldots-u_{k-\lambda+1}^+(L-1)\psi_k (u_1, \ldots, u_{k-\lambda})) \right] \]
\[ = J(u, \psi) \]

where the last equality is obtained since \( h(-y) = h(y) \) by equation (3-14) and by definition, \( L = \lambda (k-1) + 1 \).

From this fundamental result a corollary follows. If \((u^*, \psi^*)\) is an optimal strategy pair, then \((u^*, \psi^*), (\bar{u}^*, \psi^*), \text{ and } (u^*, \bar{\psi}^*)\) are also optimal strategy pairs. The proof of the first of these is based on the fact that their payoffs are the same:

\[ V = J(u^*, \psi^*) = J(\bar{u}^*, \psi^*) \]

The latter two follow from this and from the definition of a saddle-point in equation (3-5):

\[ J(u^*, \psi^*) \leq J(\bar{u}^*, \psi^*) \leq J(u^*, \bar{\psi}^*) \]

\[ V \leq J(\bar{u}^*, \psi^*) \leq V \]

and:

\[ J(u^*, \bar{\psi}^*) \leq J(u^*, \bar{\psi}^*) \leq J(u^*, \psi^*) \]

\[ V \leq J(u^*, \bar{\psi}^*) \leq V \]

and therefore both yield the Value of the game.
Indeed, any mixture of $u^*$ and $\bar{u}^*$ combined with any mixture of $\psi^*$ and $\bar{\psi}^*$ also yields the Value. By "mixture" is meant the following:
Let the evader choose his strategy by means of a random experiment whose two outcomes, $u^*$ and $\bar{u}^*$, occur with probabilities $p$ and $1-p$ respectively. Similarly let the marksman choose between $\psi^*$ and $\bar{\psi}^*$ according to the probabilities $q$ and $1-q$ respectively. The proof is obtained by writing the expected payoff:

$$E[\text{Payoff}] = pqJ(u^*, \psi^*) + (1-p)qJ(\bar{u}^*, \psi^*)$$
$$+ p(1-q)J(u^*, \bar{\psi}^*) + (1-p)(1-q)J(\bar{u}^*, \bar{\psi}^*) = V$$

and since the payoff is the Value of the game, the strategies are optimal.

It must be emphasized that these results hold for either pure strategies, i.e., $u$ and $\psi$ are a K-ary N-tuple and a set of mappings respectively, or mixed strategies, i.e., $u$ is a random variable and $\psi$ is a set of random mappings. In the latter case, no generality is lost by considering only symmetric probability distributions. This means for the evader that with no loss in generality he may restrict himself to a probability density function such that the probability of $u$ equals the probability of $\bar{u}$. Suppose for example he did not so restrict himself and obtained a density function which was not symmetric. He could then apply the previous result with $p = 1/2$ and obtain a symmetric density which also yields the Value and is therefore optimal. Thus, if an optimal solution in mixed strategies
exists then an optimal mixed strategy with a symmetric density function also exists.

7.3 The Path Function

In this section, a new function, the "path function" is defined which is a measure of the number of paths through a graph which have specified endpoints. This function arises in a surprising plethora of ways in the analysis of Austere Games. Among others, it appears in the calculation of bounds in Chapter VIII and also at several points in the analysis of finite memory games in Chapter IX.

The path function $g_k(l)$ is defined as the number of distinct paths from the origin of a k-graph to node l on the face. Since the origin corresponds to $x_0 = 0$, node l on the face corresponds to $x_k = l$, and each path corresponds to a distinct trajectory of the evader, the path function $g_k(l)$ is the number of distinct trajectories from $x_0 = 0$ to $x_k = l$. Each path in a k-graph and therefore each trajectory is formed by a specific sequence $u_1, u_2, \ldots, u_k$ of the evader's controls. A path from the origin to node l corresponding to a trajectory from $x_0 = 0$ to $x_k = l$ has, by equation (3-18), the property that:

$$u_1 + u_2 + \ldots + u_k = l.$$

A more conventional, less problem-oriented definition may be obtained by hypothesizing a number k of cells in which are placed l objects such that there are $u_i$ objects in the $i^{th}$ cell. Then:
\( g_k^K(\ell) \) is the number of ways of distributing \( \ell \) non-distinct objects in \( k \) cells such that no cell contains more than \( K-1 \) objects.

When stated in this way, \( g_k^K(\ell) \) is amenable to well-known combinatorial techniques [23]. The details of the calculation are given in Appendix B. The result is:

\[
\begin{align*}
g_k^K(\ell) &= \binom{k+\ell-1}{\ell} \quad \ell = 0, 1, \ldots, K-1 \\
g_k^K(\ell) &= \binom{k+\ell-1}{\ell} - \binom{k}{\ell} \binom{k+(\ell-K)-1}{\ell-K} \quad \ell = K, K+1, \ldots, 2K-1 \\
&\vdots \\
g_k^K(\ell) &= \sum_{m=0}^{n} (-1)^m \binom{k}{m} \binom{k+(\ell-mK)-1}{\ell-mK} \quad \ell = nK, nK+1, \ldots, (n+1)K-1 \\
&\quad n = 0, 1, 2, \ldots, K-1
\end{align*}
\]

where \( \binom{n}{m} \) denotes the binomial coefficient:

\[
\binom{n}{m} = \frac{n!}{(n-m)! \, m!}.
\]

The closed form for \( g_k^K(\ell) \) is nice to have for ascetic reasons, but it is analytically and computationally messy. Fortunately, especially from a digital computer's point of view, the path function satisfies the functional equation:

\[
\begin{equation}
\mathbf{g}_k^K(\ell) = \sum_{m=0}^{K-1} g_k^{K-1}(\ell-m) \quad (7-3a)
\end{equation}
\]

with:
\[ g_k^K(\ell) = \begin{cases} 1 & \ell = 0, 1, \ldots, K-1 \\ 0 & \text{otherwise} \end{cases} \quad (7-3b) \]

where, for convenience, \( g_k^K(\ell) \) is defined to be zero outside the range \( \ell = 0, 1, 2, \ldots, k(K-1) \). To see this, note that:

\[
g_k^K(\ell) = \text{number of paths from } x_0 = 0 \text{ to } x_k = \ell
\]

\[
= [\text{number of paths to } x_k = \ell \text{ such that } u_k = 0, \\
+ \text{number of paths to } x_k = \ell \text{ such that } u_k = 1, \\
+ \ldots + \text{number of paths to } x_k = \ell \text{ such that } u_k = K-1]
\]

\[
= [\text{number of paths from } x_0 = 0 \text{ to } x_{k-1} = \ell, \\
+ \text{number of paths to } x_{k-1} = \ell-1, + \ldots + \text{number of paths to } x_{k-1} = \ell-(K-1)]
\]

\[
[ g_{k-1}^K(\ell) + g_{k-1}^K(\ell-1) + \ldots + g_{k-1}^K(\ell-K+1) ]
\]

and, of course, there is exactly one path from the origin to each node in the first column.

A digital computer program to calculate the path function iteratively by means of equations (7-3) is easily written and the resulting algorithm is very efficient. Appendix C gives extensive tables of the path function which were calculated in this way. Note — and this can be seen either from the functional equations (7-3) or from the tables in Appendix C —— that for \( K = 2 \):

\[
g_k^2(\ell) = \binom{k}{\ell}.
\]
In order to obtain further insight into the structure of $g_K^r(k)$ and also for future explicit use in Chapter IX the following results are presented:

Symmetry Theorem:

$$g_K^r(k(K-1)-\ell) = g_K^r(\ell).$$  \hspace{1cm} (7-4)

Monotonicity Theorem:

\begin{align*}
\text{a)} \quad & \frac{g_K^r(\ell)}{g_K^r(\ell-1)} \quad \ell \leq \frac{1}{2}k(K-1) \quad (7-5a) \\
\text{b)} \quad & \frac{g_K^r(\ell)}{g_K^r(\ell)} \quad \ell = 0, 1, \ldots, k(K-1) \quad (7-5b) \\
\text{c)} \quad & \frac{g_K^r(\ell)}{g_K^r(\ell)} \quad \ell = 0, 1, \ldots, k(K-1). \quad (7-5c)
\end{align*}

The first three of these equations are easily proven by induction on $k$ while the fourth follows immediately from equation (7-3a). The details are so straightforward and the truth of the theorems so obvious from inspection of the tables in Appendix C that the proof of these theorems will be omitted.

An important corollary of equations (7-4) and (7-5a) is:

$$g_K^r_k = \max_{\ell} g_K^r(\ell) = \begin{cases} 
  g_k^r(\frac{1}{2}k(k-1)) & \text{k or K-1 even} \\
  g_k^r(\frac{1}{2}k(k-1)\pm\frac{1}{2}) & \text{both k and K-1 odd}
\end{cases}$$  \hspace{1cm} (7-6)

which says that the largest number of paths go to the node which is nearest the center of the face.
Additional results and insight may be obtained by adopting a different viewpoint. Suppose that $u_1, u_2, u_3, \ldots, u_k$ are a sequence of mutually stochastically independent random variables uniformly distributed on $U$:

$$\Pr(u_1 = a) = \frac{1}{K} \quad a = 0, 1, 2, \ldots, K-1.$$  

Then, the $K$-ary $k$-tuple $u = (u_1, u_2, \ldots, u_k)$ will be a random variable which is uniformly distributed on $U^k$:

$$\Pr(u = v) = \frac{1}{K^k} \quad v \in U^k \quad (7-7)$$

and the random variable:

$$y = u_1 + u_2 + \ldots + u_k \quad (7-8)$$

will have the probability density function:

$$\Pr(y = \ell) = \frac{K}{K^k} \ell^{K-1} \quad (7-9)$$

since the event $y = \ell$ is the union of all the distinct ways that the sum of the $u_i$ can total $\ell$ and the probability of each of these ways is given by $(7-7)$. Because of the relationship given by equation $(7-9)$ a study of the random variable $y$ can yield valuable insight into the path function.

One way of studying it is by the use of transform techniques. Define the characteristic function of the random variable $x$ to be:

$$C_x = E[z^x]$$
a function of the transform variable $z$. Now, in this specific case:

$$C_{u_i} = \frac{1}{K} + \frac{1}{K} z + \frac{1}{K} z^2 + \ldots + \frac{1}{K} z^{K-1}$$

$$= \frac{1}{K} (1 + z + z^2 + \ldots + z^{K-1})$$

$$= \frac{1}{K} \left( \frac{1-z^K}{1-z} \right)$$

while, in the more general case where the $u_i$ are not uniformly distributed but have probabilities $p_j$:

$$\Pr\{u_i = j\} = p_j \quad j = 0, 1, \ldots, K-1$$

the characteristic function is:

$$C_{u_i} = p_0 + p_1 z + p_2 z^2 + \ldots + p_{K-1} z^{K-1}$$

$$= \sum_{j=0}^{K-1} p_j z^j.$$  \hfill (7-11)

The characteristic function of the random variable $y$ in equation (7-8) is found to be:

$$C_y = E[z^y] = E[z^{u_1+u_2+\ldots+u_K}]$$

which, since the $u_i$ are mutually stochastically independent becomes:

$$C_y = E[z^{u_1}] E[z^{u_2}] E[z^{u_3}] \ldots E[z^{u_k}]$$

$$= C_{u_1} \times C_{u_2} \times C_{u_3} \times \ldots \times C_{u_k}$$

$$= (C_{u_1})^k$$

$$= \frac{1}{K} \left( \frac{1-z^K}{1-z} \right)^k.$$

\hfill (7-12)
From the definition of the characteristic function and equation (7-9):

\[ C_y = \frac{1}{k} (g^K_k(0) + g^K_k(1) z + g^K_k(2) z^2 + \ldots + g^K_k(k(1-k)) z^{k(1-k)}) \]

combined with equation (7-12) yields:

\[ c(z) = \left(1 - \frac{z^K}{1-z^K}\right)^k = \sum_{\ell=0}^{k(K-1)} g^K_k(\ell) z^\ell \quad (7-13) \]

the generating function of the path function.

The generating function yields still another method for calculating \( g^K_k(\ell) \). One simply forms \( c(z) \):

\[ c(z) = (1 + z + z^2 + \ldots + z^{K-1})^k \quad (7-14) \]

and expands it. The coefficient of \( z^\ell \) in the resulting polynomial will be \( g^K_k(\ell) \).

Letting \( z \rightarrow 1 \) in (7-14) gives:

\[ \sum_{\ell=0}^{k(K-1)} g^K_k(\ell) = k^k \]

which says that the total number of all paths from the origin of the \( k \)-graph to the face, which of course is the same as the total number of different \( K \)-ary \( k \)-tuples \( u = (u_1, u_2, \ldots, u_k) \), is \( k^k \).

The final result in this section on the path function is an asymptotic one. For large \( k \):

\[ g^K_k(\ell) \rightarrow \left(\frac{\pi}{6}\right) \left(k(2-k)^{-1}\right)^{1/2} k^k e^{-\frac{(\ell - 1/2 k(k-1))^2}{1/6 k^2(k-1)}} \quad (7-15) \]
The proof depends on the random variable \( y \) introduced by equation (7-8). The mean \( \eta_i \) and variance \( \sigma_i^2 \) of the random variable \( u_i \) which is a constituent of \( y \) are:

\[
\eta_i = \frac{1}{K} \sum_{m=0}^{K-1} m = \frac{1}{2} (K-1)
\]

\[
E[u_i^2] = \frac{1}{K} \sum_{m=0}^{K-1} m^2 = \frac{(K-1)(2K-1)}{6}
\]

\[
\sigma_i^2 = E[u_i^2] - \eta_i^2 = \frac{1}{12} (K^2 - 1)
\]

so that the mean \( \eta \) and variance \( \sigma^2 \) of \( y \) are:

\[
\eta = \frac{1}{2} k(K-1) \quad \sigma^2 = \frac{1}{12} k(K^2 - 1)
\]

Since the random variable \( y \) consists of a sum of the \( u_i \) (equation (7-8)) so that, by the Central Limit Theorem, the envelope of the density function of \( y \) in the limit approaches the Gaussian density function with \( \eta \) and \( \sigma^2 \) as above. This combined with equation (7-9) completes the proof.

7.4 Summary and Conclusions

Graph theory is a tool for the study of the structure of processes and systems. In this chapter it was used to obtain a deeper understanding of aiming and evasion games. The games were formulated as graphs and several examples were drawn and presented. These had such strong symmetry about their horizontal centerlines that a study of the symmetry properties of the games was initiated the result of which was a very useful and general symmetry theorem.
Perhaps the most useful output of the chapter though was the definition and study of the path function \( g^K(\lambda) \). The author has been continually surprised by the many and diverse ways that this function has appeared throughout his study of Austere Games. It arises in the calculation of bounds on the Value in Chapter VIII and in the analysis of finite memory games in Chapter IX. Although it is apparently not a component of the exact analytical solution for \( K = \lambda = 2 \) given in Chapter X, if experience is any guide it will reappear when the solution is extended to arbitrary \( K \) and \( \lambda \). For example, an innocuous looking 1/2 in the solution for \( K = \lambda = 2 \) might become \( 1/g^K(1) \) in its extension to arbitrary \( K \) and \( \lambda \).

This is speculation of course. However, the path function has proven itself to be both useful and ubiquitous and for that reason has been studied in detail in this chapter: a closed form expression for it has been derived, an algorithm for calculating it presented, tables of it given (in Appendix C), the generating function for it obtained, theorems describing its general symmetry and structure proven and its asymptotic form explained. Although many of these results are explicitly used in the sequel, the main purpose of section 7.3 which discusses the path function is to obtain a basic understanding of this useful function and its properties.
CHAPTER VIII
BOUNDS ON THE VALUE

In this chapter a set of upper and lower bounds on the Value of the game is developed. As explained in section 8.1, such bounds are necessary if one is to effectively evaluate the worth of any proposed approximate optimal strategy. Section 8.2 gives a technique based on saddle-point theory for obtaining bounds on the Value and actual numerical bounds are derived in section 8.3 using this technique. These are depicted in figures and tables and their behavior as K, λ and N are varied is discussed. The chapter closes with the summary and conclusions in section 8.4.

8.1 Rationale of Bounds on the Value

In a situation in which finding the exact optimal solution of the game is very difficult, the natural inclination is to consider and look for approximations to the marksman's and the evader's optimal strategies. This is especially pertinent in view of the well-known effect which Isaacs [10] calls "the principle of flat laxity". This principle simply states that when attempting to maximize (or minimize) a function "the value of the argument at a maximum (or minimum) is generally not critical" because for many types of maxima (or minima) the first derivatives of the function at the maximum are zero, the
function is "flat", and slight deviations from optimal behavior are not heavily penalized.

This authoritative pronouncement may be applied to saddle-points and so to game theory. However, when attempting to develop approximations in game theory, one must be more than a little wary of the techniques of such subjects as Optimal Control Theory or Mathematical Physics. For in game theory, conflict is the essence and there is always an opponent ready to exploit to the uttermost any deviation from the optimal. As the whole theory is based on the assumption that the opponent is intelligent, it must be assumed that he will act rationally and pounce on any weak points in his adversary's defenses.

An example from reference [10] may serve to focus the idea. If one player is defending a city with a hundred gates and his strategy consists of an allocation of his forces amongst them, he will have a close approximation to an excellent strategy if he adequately guards ninety-nine. But, if his opponent's intelligence is good, all his forces will surge through the neglected gate and the near perfect defense is useless.

Obviously then, any proposed approximate strategy must be viewed not in the shade of conventional approximation theory but in the light of the min-max viewpoint. It must be examined in terms of the worst advantage the opponent can take of it.

In developing such approximations it is very useful, if not essential, to have an idea of the actual Value of the game so that
any deviations from optimality can be evaluated quantitatively. For this purpose, the exact Value is preferable but, failing this, the firm knowledge that the Value lies within a narrow and specified range is almost as beneficial.

8.2 A Technique for Obtaining Bounds

And so one is led to various attempts to specify in terms of upper and lower bounds a range within which the Value must lie. The basis for these attempts is rooted in equation (3-6):

\[ V = \min_{\phi} \max_{\psi} J(\phi, \psi) = \max_{\psi} \min_{\phi} J(\phi, \psi). \]  

(8-1)

To obtain an upper bound, one may specify some strategy, \( \tilde{\phi} \) say, for the evader so that:

\[ V = \min_{\phi} \max_{\psi} J(\phi, \psi) \leq \max_{\psi} J(\tilde{\phi}, \psi). \]  

(8-2)

Alternately, one may specify a class, \( \tilde{\psi} \) say, of strategies for the evader which is a subset of the set of all possible \( \psi \) and then allow the evader to find his best possible strategy within that class so that:

\[ V = \min_{\phi} \max_{\psi} J(\phi, \psi) \leq \min_{\phi \in \tilde{\psi}} \max_{\psi} J(\phi, \psi). \]  

(8-3)

If the specified strategy \( \tilde{\phi} \) is auspiciously chosen or the class \( \tilde{\psi} \) is a wide one, these upper bounds may be expected to be "tight" — not far above \( V \) — and the corresponding strategy for the evader (but not necessarily for the marksman) is a good approximation. The
reverse is also true: If the specified strategy or class of strategies is poorly selected, the bounds in (8-2) and (8-3) will be loose.

Similar techniques exist for the lower bound. If the marksman's strategy \( \tilde{\psi} \) is specified, then:

\[
\min_{\phi} J(\phi, \tilde{\psi}) \leq \max_{\psi} \min_{\phi} J(\phi, \psi) = V \tag{8-4}
\]

or if a class \( \bar{\psi} \) of strategies for the marksman is specified within which the marksman is free to optimize his performance then:

\[
\max_{\psi \in \bar{\psi}} \min_{\phi} J(\phi, \psi) \leq \max_{\psi} \min_{\phi} J(\phi, \psi) = V . \tag{8-5}
\]

As before, if the choice of \( \tilde{\psi} \) or \( \bar{\psi} \) is a good one, the bounds will be tight and the resulting strategies for the marksman (but not necessarily for the evader) will be good approximations. The converse is also true.

8.3 Bounds on the Value

To exemplify the technique, the following bounds are presented:

\[
\frac{S}{L} \leq V \leq S G^K K^{-\lambda} . \tag{8-6}
\]

Proof of the lower bound: If at each stage the marksman chooses his aiming point randomly with each of the \( L = \lambda(K-1) + 1 \) possibilities equally likely, there will be \( S \) shots with the probability of a hit on each equal to \( \frac{1}{L} \) so that the payoff is \( \frac{S}{L} \), independent of what the evader does. This payoff is, by equation (8-4), a lower bound on \( V \).
Proof of the upper bound: If at each stage the evader chooses his control $u_k$ randomly with each of the $K$ possibilities equally likely, then his position $z_k$ will be a random variable:

$$z_k = u_k - \lambda + 1 + u_{k-\lambda + 2} + \ldots + u_k$$

whose probability density function is given by equation (7-9):

$$\Pr(z_k = \ell) = \frac{K}{\lambda^\ell} K^{-\lambda}.$$  \hspace{1cm} (8-7)

By equation (8-2) an upper bound is obtained by allowing the marksman to take full advantage of the situation by maximizing his hit probability per shot and thus maximizing the expected number of hits. To maximize his hit probability, he should choose $\hat{z}_k$ to be the value of $\ell$ which maximizes equation (8-7). By equation (7-9) this yields:

$$\Pr\{\text{hit at time } k\} = \max_{\ell} \frac{K}{\lambda^\ell} K^{-\lambda} = \frac{K}{\lambda} K^{-\lambda}$$

and the upper bound in (8-6) follows, completing the proof.

Although the calculation of these bounds is not difficult, they furnish valuable insight into the magnitude of $V$ and its variation with $K$ and $\lambda$. Table 8-1 gives values of these bounds for a number of different combinations of $K$ and $\lambda$. They are also plotted in Figures 8-1 and 8-2. Notice that the quantity $V/S$, the optimal average hit probability per shot, is tabulated and plotted. This is intended to normalize the bounds with respect to the length of the game. Clearly for games of long duration, doubling the length
Figure 8-1  Bounds on the Value when $\lambda = 2$. 
Figure 8-2  Bounds on the Value when $K = 2$. 
Table 8-1  Upper and lower bounds on the Value of the game.

This table gives pairs of lower and upper bounds both exactly as fractions and approximately as decimals on V/S, the optimal hit probability per shot, of the Austere Game in which the evader's control may take on any of K values and the time delay is $\lambda$.

<table>
<thead>
<tr>
<th>K</th>
<th>2</th>
<th>3</th>
<th>$\lambda$</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>($\frac{1}{3}, \frac{2}{4}$)</td>
<td>($\frac{1}{4}, \frac{3}{8}$)</td>
<td>($\frac{1}{5}, \frac{6}{16}$)</td>
<td>($\frac{1}{6}, \frac{10}{32}$)</td>
<td>($\frac{1}{7}, \frac{20}{64}$)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>($\frac{1}{5}, \frac{3}{9}$)</td>
<td>($\frac{1}{7}, \frac{7}{27}$)</td>
<td>($\frac{1}{9}, \frac{19}{81}$)</td>
<td>($\frac{1}{11}, \frac{51}{243}$)</td>
<td>($\frac{1}{13}, \frac{141}{729}$)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>($\frac{1}{7}, \frac{4}{16}$)</td>
<td>($\frac{1}{10}, \frac{12}{64}$)</td>
<td>($\frac{1}{13}, \frac{44}{256}$)</td>
<td>($\frac{1}{16}, \frac{155}{1024}$)</td>
<td>($\frac{1}{19}, \frac{580}{4096}$)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>($\frac{1}{9}, \frac{5}{25}$)</td>
<td>($\frac{1}{13}, \frac{19}{125}$)</td>
<td>($\frac{1}{17}, \frac{85}{625}$)</td>
<td>($\frac{1}{21}, \frac{381}{3125}$)</td>
<td>($\frac{1}{25}, \frac{1751}{15625}$)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>K</th>
<th>2</th>
<th>3</th>
<th>$\lambda$</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>(.3333,.5000)</td>
<td>(.2500,.3750)</td>
<td>(.2000,.3750)</td>
<td>(.1666,.3125)</td>
<td>(.1429,.3125)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>(.2000,.3333)</td>
<td>(.1429,.2593)</td>
<td>(.1111,.2346)</td>
<td>(.0909,.2099)</td>
<td>(.0769,.1934)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>(.1429,.2500)</td>
<td>(.1000,.1875)</td>
<td>(.0769,.1719)</td>
<td>(.0625,.1514)</td>
<td>(.0526,.1416)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>(.1111,.2000)</td>
<td>(.0769,.1520)</td>
<td>(.0588,.1360)</td>
<td>(.0476,.1219)</td>
<td>(.0400,.1121)</td>
<td></td>
</tr>
</tbody>
</table>
of the game should approximately double the expected number of hits so that \( V \) is, at least in the limit, proportional to \( S \), the number of shots. The quantity \( V/S \) is therefore a reasonable measure of the goodness of an aiming system.

Notice also that, although the upper and lower bounds in Figures 8-1 and 8-2 are not too far apart, they seem to stay approximately parallel and do not show much inclination to converge. This conclusion is supported by the calculation of the ratio of the upper bound to the lower bound:

\[
\frac{\text{upper bound}}{\text{lower bound}} = \frac{\lambda (K-1)+1}{\sqrt{2\pi^2 \lambda (K^2 - 1)}} = \sqrt{\frac{6\lambda}{\pi}}
\]

obtained with aid of asymptotic formula (7-15). Thus, the two stay roughly the same percentage apart if \( \lambda \) is held constant while \( K \) increases, but they actually diverge if \( K \) is held constant and \( \lambda \) is allowed to increase.

A tighter upper bound for the case \( K = \lambda = 2 \) may be obtained by restricting the evader to the use of the strategy given by Isaacs for Isaacs' Game and applying equation (8-2). This strategy is described in equations (3-22) and yields a payoff per shot:

\[
\frac{1}{2}(3 - \sqrt{5}) \approx .382
\]

no matter what strategy is chosen by the marksman. It must be remembered that this is not the Value of the game and is truly only a somewhat tighter upper bound.
8.4 Summary

This chapter has presented a technique for obtaining bounds on the Value of the game and used it to calculate an upper and a lower bound. These were displayed in Figures 8-1 and 8-2 and in Table 8-1.

The value of such bounds is that they enable one to get a feel for the numerical size of V and for its dependence on the parameters K, λ, and N. Not surprisingly, it was found that V, the expected number of hits when both participants play optimally, is roughly proportional to S = N - λ + 1, the total number of shots and thus V/S, the ratio of hits to shots, is an important parameter. The dependence of V on K and λ was shown to be more complicated but it may at least be said that both upper and lower bounds decrease monotonically with K and λ.

The bounds also have the advantage that they may be used to estimate the Value V and thus furnish a standard by means of which various approximately optimal strategies can be evaluated. This advantage should not be underestimated. Several times during the course of this work, the author or one of his colleagues has come up with an elegant and ingenious strategy for one or the other of the players which was allegedly "near-optimal". Such inspired guesses could usually be quickly and efficiently tested by comparing their payoff to V estimated from the bounds. Unfortunately, most were summarily rejected as a result. However, the worth of the bounds was proven.
CHAPTER IX

FINITE MEMORY GAMES

It has been pointed out that part of the difficulty of these games arises from the fact that the aiming points of the marksman and the controls of the evader apparently must depend upon the whole past trajectory. The domains of the evader's closed-loop control functions \( \phi_k \) in (3-15) and the marksman's aiming point functions \( \psi_k \) in (3-16) reflect this difficulty by including all past controls and measurements. This phenomenon makes it very difficult to apply any of the more-or-less routine methods of optimization. Most such techniques fail completely when one attempts to apply them to functions whose domains are of very high dimensionality.

In such a situation, physical intuition suggests that the dependence of the marksman's aiming points and the evader's controls upon measurements made in the far past will be slight. It seems likely therefore that neglecting the far past when calculating strategies will cause little performance degradation for either. Thus a reasonable technique for obtaining approximate strategies might be to restrict the functions \( \phi_k \) and \( \psi_k \) to depend only upon a limited number of the controls from the immediate past and allow the participants to optimize their performance within this restriction. Presumably this approach will never yield
optimal strategies and the Value but it may be expected to yield
tighter bounds on it and good approximate strategies.

Games with such restrictions, called finite memory games, are
the subject of this chapter. Section 9.1 defines the general forms
of such games and derives a number of their properties. Among the
latter perhaps the most important is that their solutions converge
to the Value of the original Austere Game.

The remainder of the chapter is devoted to the analysis of several
of the simpler examples of finite memory games. Section 9.2 analyzes
in depth the case in which the marksman is allowed zero memory and
presents the Value and the optimal strategies exactly in closed form.
These are illustrated by an example in section 9.3. Section 9.4
describes the game in which the evader is allowed zero memory and
derives an approximate technique for its solution. An example of this
technique is given in section 9.5. The next level of difficulty is
reached in section 9.6 which presents an analysis of the game in which
the evader may remember no more than one of his past controls. The
Markovian strategies obtained for this game are good ones to use in
the original game since they yield a payoff reasonably close to V. The
chapter terminates with the summary and conclusions in section 9.7.

9.1 Finite Memory Games

In this section, the definition of the general finite memory game
is given and some of its properties are derived. One of the most useful
of these is that if one player's memory of the past is limited, then his opponent may limit the dependence of his own strategy upon the distant past without suffering any performance degradation. Also, it will be shown that as the amount of memory is increased, the sequence formed by the solutions of finite memory games converges monotonically to the Value of the original game.

The term finite memory game is used here to mean the aiming and evasion game in Austere form in which one or the other of the participants is restricted to the use of only a limited number of the most recent of his past measurements when making his decisions. In other words, either the marksman or the evader is allowed only a finite memory of specified size so that every time he makes a new measurement he must "forget" the measurement made farthest back in the past in order to have room in his memory to store the latest one.

In particular, define the ER-Game to be the Austere Game in which the evader is restricted to the use of only his \( R \) most recent past controls in his closed-loop strategy computations. Specifically, the closed-loop pure strategy \( \phi = \{\phi_1, \phi_2, \ldots, \phi_N\} \) is restricted to the form:

\[
\begin{align*}
\phi_1 &= \phi_1 \\
\phi_2(u_1) &= \phi_2(u_1) \\
\phi_3(u_1, u_2) &= \phi_3(u_1, u_2) \\
\vdots
\end{align*}
\]
\[ u_{R+1} = \phi_{R+1}(u_1, u_2, \ldots, u_R) \]
\[ u_{R+2} = \phi_{R+2}(u_2, u_3, \ldots, u_{R+1}) \]
\[ u_{R+3} = \phi_{R+3}(u_3, u_4, \ldots, u_{R+2}) \]
\[ \vdots \]
\[ u_N = \phi_N(u_{N-R}, u_{N-R+1}, \ldots, u_{N-1}) \]

The set of all such strategies will be denoted by \( \Phi_R \). Within the framework of these strategies the evader optimizes his performance. The marksman has no restrictions on his strategy \( \psi_k \) and it must be assumed that he will take full advantage of the evader's handicap to increase his payoff \( J \).

Although there are no official restrictions on the marksman, a very useful feature of the ER-Game is that he need only remember his \( R \) most recent measurements. This happens because the marksman must estimate:

\[ z_k = u_{k-\lambda+1} + u_{k-\lambda+2} + \ldots + u_k \]

which by equations (9-1) is only a function of \( u_{k-\lambda-R+1}, u_{k-\lambda-R+2}, \ldots, u_{k-\lambda}, u_{k-\lambda+1}, \ldots, u_k \) and only the first \( R \) of these are known to the marksman. Thus, in the ER-Game, the marksman can, with no loss of generality, be restricted to the strategies:
\[ \hat{z}_1 = \psi_1 \]
\[ \hat{z}_{\lambda + 1} = \psi_{\lambda + 1}(u_1) \]
\[ \hat{z}_{\lambda + 2} = \psi_{\lambda + 2}(u_1, u_2) \]
\[ \vdots \]
\[ \hat{z}_{\lambda + R} = \psi_{\lambda + R}(u_1, u_2, \ldots, u_R) \]
\[ \hat{z}_{\lambda + R + 1} = \psi_{\lambda + R + 1}(u_2, u_3, \ldots, u_{R + 1}) \]
\[ \hat{z}_{\lambda + R + 2} = \psi_{\lambda + R + 2}(u_3, u_4, \ldots, u_{R + 2}) \]
\[ \vdots \]
\[ \hat{z}_N = \psi_N(u_{N - \lambda - R + 1}, u_{N - \lambda - R + 2}, \ldots, u_{N - \lambda}) \].

The set of all such strategies will be denoted by \( \Psi_R \).

The Value of the ER-Game is:

\[ W_{ER} = \min_{\phi \in \Phi_R} \max_{\psi \in \Psi_R} J(\phi, \psi) = \min_{\phi \in \Phi_R} \max_{\psi \in \Psi_R} J(\phi, \psi) \]

which, by equation (8-3), is an upper bound on the Value of the original game:

\[ V \leq W_{ER}. \]

It must be kept in mind that the object of this study of finite memory games is to obtain good approximations to the Value and strategies of the original Austere Game. It is vitally important, therefore, to understand exactly the relationship of \( W_{ER} \), the value of the finite
memory ER-Game to $V$ the Value of the original game. The following result pinpoints this relationship.

The sequence formed by the solutions of the ER-Game as $R$ increases converges monotonically to the Value of the original game:

$$V < \ldots < W_2 \leq W_1 \leq W_0.$$  \hspace{1cm} (9-4)

The proof is based on the fact that:

$$\phi_0 \subseteq \phi_1 \subseteq \phi_2 \subseteq \ldots \subseteq \phi_{N-1} = \phi.$$

Thus, in the $E(R+1)$-Game, the evader can always do at least as well as he did in the ER-Game because the set of strategies $\phi_{R+1}$ of the former includes as a proper subset the set of strategies $\phi_R$ of the latter, so that he can reduce the payoff to no more than $W_{ER}$ by simply employing in the $E(R+1)$-Game the same strategy he used in the ER-Game. Hence:

$$W_{Er(R+1)} \leq W_{ER}.$$

The final inequality in (9-4) results from (9-3). The series converges to $V$ because the $E(N-1)$-Game is identical to the original game (cf. equations (9-1) and (3-15) for $R = N - 1$) so that:

$$W_{Er(N-1)} = V$$

which completes the proof.

A quite similar set of results may be obtained by restricting the marksman rather than the evader. Let the MR-Game be the Austere Game
in which the marksman is restricted to the use of only his R most recent measurements in his aiming point calculations. Specifically, the closed-loop pure strategy \( \psi = \{\psi_\lambda, \psi_{\lambda+1}, \ldots, \psi_N\} \) is restricted to the form given in equations (9-2) so that \( \psi \in \Psi_R \). Within this limitation, the marksman is free to optimize his performance. The evader has no restrictions on his strategy \( \phi \) and it must be assumed that he will take full advantage of the marksman's handicap to decrease \( J \).

Although there are no official restrictions on the evader, a very useful feature of the MR-Game is that he need only remember his \( R + \lambda - 1 \) most recent controls. This is because at time \( k \) the evader, in order to optimize his performance, need only know where he is presently (with respect to the moving origin \( x_{k-\lambda} \)) which requires knowledge of \( u_{k-\lambda+1}, u_{k-\lambda+2}, \ldots, u_{k-1} \), and where the marksman plans to shoot which can be calculated from knowledge of \( u_{k-\lambda-R+1}, u_{k-\lambda-R+2}, \ldots, u_{k-\lambda} \). Therefore, the sum total of the evader's required knowledge is no more than \( \lambda + R - 1 \) of his past controls and with no loss of generality he may restrict his search for an optimal strategy to the set \( \Phi_{R+\lambda-1} \).

The Value of the MR-Game is:

\[
W_{\text{MR}} = \max_{\psi \in \Psi_R} \min_{\phi \in \Phi_{\lambda+R-1}} J(\phi, \psi) = \max_{\phi \in \Phi_{R+\lambda-1}} \min_{\psi \in \Psi_R} J(\phi, \psi)
\]

which by equation (8-5) is a lower bound on the Value of the original game:

\[
W_{\text{MR}} \leq V. \quad (9-5)
\]
To clarify the relationship of $W_{MR}$ to $V$, the following important result is presented. The sequence formed by the solutions of the MR-Game as $R$ increases converges monotonically to the Value of the original game:

$$W_{M0} \leq W_{M1} \leq W_{M2} \leq \cdots \leq V.$$  \hspace{1cm} (9-6)

The proof is based on the fact that:

$$\Psi_0 \subseteq \Psi_1 \subseteq \Psi_2 \subseteq \cdots \subseteq \Psi_{N-\lambda} = \Psi.$$

Therefore, in the $M(R+1)$-Game, the marksman can always do at least as well as in the MR-Game because his set of strategies $\Psi_{R+1}$ includes $\Psi_R$ as a proper subset enabling him to obtain a payoff of at least $W_{MR}$ by simply employing in the $M(R+1)$-Game the same strategy which he used in the MR-Game. Hence:

$$W_{MR} \leq W_{MR+1}.$$  \hspace{1cm}

The final inequality in (9-6) results from (9-5). The series converges to $V$ because the $M(N-\lambda)$-Game is identical to the original game (cf. equations (9-2) and (3-16) with $R = N - \lambda$) so that:

$$W_{M(N-\lambda)} = V$$

which completes the proof.
9.2 Solution of the MO-Game

In the MO-Game the marksman is allowed no memory (actually one word of memory since it is tacitly assumed that he is able to remember $x_{k-\lambda}$) so that each of his shots is independent of his past ones. Since he will need a mixed strategy, his aiming point $\hat{z}_k$ will be a stochastically independent random variable and his problem is to choose its probability density. The evader however must in general choose a closed-loop mixed strategy which is a function of his $\lambda$ past controls. This section analyzes the MO-Game and presents both the evader's and marksman's strategies in closed form.

The evader's desire is to minimize the probability of a hit at each time $k$. The marksman always directs his fire to maximize the hit probability by shooting at the most probable location of the evader so that:

$$P_r\{\text{hit at time } k\} = \max_{\ell} P_r\{z_k = \ell\}$$

$$= \max_{\ell} P_r\{u_{k-\lambda+1} + \ldots + u_k = \ell\}.$$  \hspace{1cm} (9-7)

The evader would like to choose his controls $u_{k-\lambda+1}, u_{k-\lambda+2}, \ldots, u_k$ to minimize this probability.

What is the best the evader can do? Since the sum over $\ell$ must be one, the best he can do is to choose his $u_{\hat{i}}$ so that:

$$P_r\{z_k = \ell\} = \frac{1}{L} \quad \ell = 0, 1, \ldots, L-1.$$ \hspace{1cm} (9-8)
Since \( z_k = u_{k-\lambda+1} + u_{k-\lambda+2} + \ldots + u_k \) and since there are \( g^K_k(\ell) \) different ways for this sum to total \( \ell \), the joint probability density function of the \( u_1 \) must be:

\[
Pr\{u_1 = a_1; i = k-\lambda+1, \ldots, k\} = \frac{1}{Lg^K_\lambda(a_k+a_{k-1}+\ldots+a_{k-\lambda+1})}. \quad (9-9)
\]

If this is done, the joint probability of the most recent \( \lambda-1 \) controls is:

\[
Pr\{u_1 = a_1; i = k-\lambda+1, \ldots, k-1\} = \sum_{i=0}^{K-1} \frac{1}{Lg^K_\lambda(i+a_{k-1}^++\ldots+a_{k-\lambda+1}^+)} \quad \text{(9-10)}
\]

while the conditional probability of \( u_k \) given the rest is:

\[
Pr\{u_k = a_k | u_1 = a_1; i = k-\lambda+1, \ldots, k-1\} = \frac{1}{g^K_\lambda(a_k^+\ldots+a_{k-\lambda+1}^+)} \sum_{i=0}^{K-1} \frac{1}{Lg^K_\lambda(i+a_{k-1}^+\ldots+a_{k-\lambda+1}^+)} \quad \text{(9-11)}
\]

The optimal mixed strategy of the evader may now be specified. At each time point \( k \), the evader should choose \( u_k \) by means of a random experiment whose outcomes have the probability density function given by (9-11). Notice that this probability density function is conditioned on his \( \lambda-1 \) most recent controls as it should be. The evader should choose his first \( \lambda-1 \) controls to have the joint probability:

\[
Pr\{u_1 = a_1; i=1,2,\ldots,\lambda-1\} = \frac{1}{L} \sum_{i=0}^{K-1} \frac{1}{g^K_\lambda(i+a_1^+\ldots+a_{\lambda-1}^+)} \quad \text{(9-12)}
\]
If the evader follows the mixed strategy given by (9-11) and (9-12), then (9-10) holds for all k. The proof is by induction: For k = \lambda, the assertion is true by (9-12). Assume it is true for k = m. Then the joint density function of \( u_{m-\lambda+1}, u_{m-\lambda+2}, \ldots, u_m \) is given by (9-9) with k = m and summing (9-9) over a \( m-\lambda+1 \) yields:

\[
\Pr\{u_i = a_i; i=m-\lambda+2, m\} = \frac{1}{L} \sum_{j=0}^{K-1} \frac{1}{g_\lambda(a_{m-\lambda+2} + \ldots + a_{m-\lambda+2} + j)}
\]

which shows the assertion is true for k = m+1 and completes the proof.

That this mixed strategy is indeed optimal will now be shown. If the evader persists in this policy, then equations (9-7) and (9-8) show that the payoff will be \( \frac{1}{L} \) per shot for a total payoff of \( \frac{S}{L} \). With this particular mixed strategy (equations (9-11) and (9-12)) the evader is assured that the payoff will be no more than \( \frac{S}{L} \); there might be a different mixed strategy for which the payoff is lower but the evader need never accept a higher payoff. This is written:

\[
W_{M0} \leq \frac{S}{L}
\]

But the lower bound in equation (8-6) was derived without any assumptions about memory so that:

\[
\frac{S}{L} \leq W_{M0}
\]

and therefore:

\[
W_{M0} = \frac{S}{L}
\]
The marksman's optimal strategy is to choose his aiming point $z_k$ to be a random variable uniformly distributed over $z$:

$$\Pr\{z_k = \ell\} = \frac{1}{L}, \quad \ell = 0, 1, 2, \ldots, L-1$$

which, no matter what strategy the evader uses, yields a hit probability per shot of $1/L$ and a total payoff of $S/L$ which is the Value.

9.3 Example

As an example of the MO-Game, consider the case where $K = 2$ and $\lambda = 3$ so that $L = \lambda(K-1) + 1 = 4$. The path function from Table C-1 in Appendix C is:

$$g_3^2(0) = 1, \quad g_3^2(1) = 3, \quad g_3^2(2) = 3, \quad g_3^2(3) = 1$$

so that the evader's optimal mixed strategy is:

$$\Pr\{u_k = 0 \mid u_{k-1} = 0, u_{k-2} = 0\} = \frac{3}{4},$$

$$\Pr\{u_k = 1 \mid u_{k-1} = 0, u_{k-2} = 0\} = \frac{1}{4},$$

$$\Pr\{u_k = 0 \mid u_{k-1} = 1, u_{k-2} = 0\} = \frac{1}{2},$$

$$\Pr\{u_k = 1 \mid u_{k-1} = 1, u_{k-2} = 0\} = \frac{1}{2},$$

$$\Pr\{u_k = 0 \mid u_{k-1} = 0, u_{k-2} = 1\} = \frac{1}{4},$$

$$\Pr\{u_k = 1 \mid u_{k-1} = 0, u_{k-2} = 1\} = \frac{1}{2},$$

$$\Pr\{u_k = 0 \mid u_{k-1} = 1, u_{k-2} = 1\} = \frac{1}{2},$$

$$\Pr\{u_k = 1 \mid u_{k-1} = 1, u_{k-2} = 1\} = \frac{3}{4}.$$

This is most easily illustrated on the 3-graph of the game as shown in Figure 9-1.
Figure 9-1 Example of evader's optimal mixed strategy in MD-Game.

Notice that this optimal mixed strategy opposes the tendency of the probability mass to concentrate in the center of the face (as described by the Central Limit Theorem) by investing in the evader a tendency to move toward the outside. If the evader is dead center he must move toward the outside. Once off center there is a 3 to 1 bias in his probabilities to stay off center.

The probabilities of reaching the various nodes in the column immediately to the left of the face may be calculated with the aid of equation (9-10):

\[
\begin{align*}
\Pr(\text{node 0}) &= 1/3 \\
\Pr(\text{node 1}) &= 1/3 \\
\Pr(\text{node 2}) &= 1/3
\end{align*}
\]

the uniform distribution being purely coincidental. Using these, the
probabilities of reaching the various nodes on the face may be calculated and, as expected, they all turn out to be \( \frac{1}{L} = \frac{1}{4} \) as shown in Figure 9-1.

9.4 Solution of the EO-Game

In the EO-Game, the evader is allowed no memory of his past controls --- he must "forget" each immediately after he uses it. The result is that he must choose each control independently. If he uses a mixed strategy so that \( u_1, u_2, \ldots, u_N \) is a sequence of mutually independent random variables, his problem is to choose their probability density function.

Assume that his choice is \( p_0, p_1, \ldots, p_{K-1} \) where:

\[
p_i = \Pr\{u_k = i\} \quad i = 0, 1, \ldots, K-1.
\]

Then \( z_k \):

\[
z_k = u_{k-\lambda+1} + u_{k-\lambda+2} + \ldots + u_k
\]

will be a random variable whose probability density function:

\[
q_\ell = \Pr\{z_k = \ell\} \quad \ell = 0, 1, 2, \ldots, L-1
\]

is related to the \( p_i \) by:

\[
q_\ell = p_{i_1} \times p_{i_2} \times p_{i_3} \times \ldots \times p_{i_\lambda}
\]

where the summation is over all \( i_1, i_2, \ldots, i_\lambda \) such that:
The marksman, by the results obtained in section 9.1, need not choose his aiming point \( \hat{z}_k \) to be a function of any of his measurements. In fact, since he desires to maximize his hit probability he should choose \( \hat{z}_k = \lambda^* \) where \( \lambda^* \) is the value of \( \lambda \) which maximizes \( q_{\lambda}^* \:

\[
q_{\lambda}^* = \max_{\lambda} q_{\lambda} \tag{9-17}
\]

in which case \( q_{\lambda}^* \) is the hit probability and \( S q_{\lambda}^* \) the payoff. The evader must choose the set of \( p_i, i = 0, 1, 2, \ldots, K-1 \) to yield the minimum payoff which is the Value of the game:

\[
W_{EO} = \min_{p_i} \max_{\lambda} S q_{\lambda} = S \min_{p_i} q_{\lambda}^* \tag{9-18}
\]

But what is the set of \( p_i \) which minimizes \( q_{\lambda}^* \)? Finding it would be a rather straightforward calculus problem were it not for the extremely complicated form of equation (9-16). In the following pages, an exact solution is obtained for the case \( \lambda = 2 \) and, for the general case, an approximate but fairly accurate solution is developed.

Case A: \( \lambda = 2 \):

In the case of \( \lambda = 2 \), formula (9-16) is not too difficult to expand and evaluate:

\[
q_{\lambda} = \sum_{i=0}^{\lambda} \ p_i \ p_{\lambda-1} \quad \lambda \leq K-1
\]

\[
q_{\lambda} = \sum_{i=\lambda}^{K-1} \ p_i \ p_{\lambda-1} \quad \lambda \geq K-1
\]
Now it will be shown that $\xi^* = K-1$, i.e., the node of maximum probability occurs at the center of the face of the 2-graph. This follows from the result, derived in Chapter VII, that no loss of generality is incurred by assuming the density function of $u_1$ to be symmetric:

$$p_i = p_{K-1-i}$$

so that:

$$q_k = \sum_{i=0}^{k} p_i p_{k-i} \leq \sum_{i=0}^{k} p_i^2 \leq \sum_{i=0}^{K-1} p_i^2 = \sum_{i=0}^{K-1} p_i p_{K-1-i} = q_{K-1}$$

where the first inequality is Schwarz's Inequality. And, comparing this to equation (9-17) which defines $\xi^*$, it is found that $\xi^* = K-1$.

Now the problem as outlined in (9-18) is:

$$W_{E_0} = S \min \sum_{i=0}^{K-1} p_i^2$$

subject to:

$$\sum_{i=0}^{K-1} p_i = 1$$

$$p_i \geq 0.$$  \hspace{1cm} (9-19a)  \hspace{1cm} (9-19b)  \hspace{1cm} (9-19c)

The solution obtained by elementary calculus is:

$$p_i = \frac{1}{K} \quad i = 0, 1, \ldots, K-1$$

and:

$$W_{E_0} = \frac{1}{K}.$$
In other words, the best the evader can do in the EO-Game with \( \lambda = 2 \) is to choose his controls randomly with each element of \( U \) equally likely.

This is exactly what intuition predicts. One instinctively expects that the evader's best performance comes when he spreads his probability distribution out as uniformly as possible. Indeed this maximizes the entropy of the evader's controls (in Shannon's sense [24]) which, it might be expected, should optimize his performance. Unfortunately, this sort of physical reasoning fails for \( \lambda > 2 \) as will now be shown.

Case B: \( \lambda > 2 \):

The case of general \( \lambda \) presents a nearly insurmountable analytic problem. The difficulty arises because, to find \( W_{E0} \) in equation (9-18), one must first find the maximum over \( \ell \) of \( q_k \) which, because of the complexity of equation (9-16), is an extremely arduous task. Moreover, one must then find the minimum over \( p_0, p_1, \ldots, p_{K-1} \) of the result.

To bypass the difficulty, an approximate method based on linearizing equation (9-16) about a set of \( p_i \) near the minimum will be described.

The characteristic function of \( u_i \) as given in equation (7-11) is:

\[
C_{u_i} = p_0 + p_1 z + p_2 z^2 + \ldots + p_{K-1} z^{K-1}
\]

and the characteristic function of \( z_k \) is:

\[
C_{z_k} = q_0 + q_1 z + q_2 z^2 + \ldots + q_{L-1} z^{L-1}
\]

(9-20)
and the two are related by (cf. equation (7-12)):

\[ C_{z, k} = C_{u_1} \times C_{u_2} \times C_{u_3} \times \ldots \times C_{u_\lambda} = (p_0 + p_1 z + p_2 z^2 + \ldots + p_{k-1} z^{k-1})^\lambda. \tag{9-21} \]

Now, the physical reasoning described in Case A leads one to suspect that the optimal distribution will be nearly uniform over the \( K \) elements of \( U \). In view of this, let:

\[ p = \frac{1}{K} + \delta_i \quad i = 0, 1, \ldots, K-1 \tag{9-22} \]

where the \( \delta_i \) are expected to be "small" and in any case must satisfy:

\[ \sum_{i=0}^{K-1} \delta_i = 0 \tag{9-23} \]

for the sum of the \( p_i \) to be one. Substituting (9-22) into (9-21) and retaining only the first order term in the \( \delta_i \) yields:

\[ C_{z, k} = \frac{1}{K^\lambda} (1 + z + z^2 + \ldots + z^{K-1})^\lambda \]

\[ + \frac{\lambda}{K^{\lambda-1}} (1 + z + z^2 + \ldots + z^{K-1})^{\lambda-1} (\delta_0 + \delta_1 z + \ldots + \delta_{K-1} z^{K-1}). \]

And, recognizing the generating function of the path function in equations (7-13) and (7-14), one obtains:

\[ C_{z, k} = \frac{1}{K^\lambda} \left( \sum_{\ell=0}^{\lambda-1} \ell K^{\lambda-1} z^{\ell} \right) + \frac{\lambda}{K^{\lambda-1}} \left( \sum_{\ell=0}^{\lambda-1} \ell K^{\lambda-1} z^{\ell} \right) \left( \sum_{i=0}^{K-1} \delta_i z^i \right). \]
Equating like powers of \( z \), here and in (9-20), gives a linear approximation for \( q_\lambda \):

\[
q_\lambda = \frac{1}{K\lambda} s^K(\lambda) + \frac{\lambda}{K^{\lambda-1}} \sum_{i=0}^{K-1} e^K_{\lambda-1}(\lambda-i)\delta_i
\]  

valid for "small" \( \delta_i \) which satisfy (9-23).

The evader's problem in equation (9-18) may now be restated as:

\[
W_{EO} = \min_{\delta_i} \max_{\lambda} \sum_{i=0}^{K-1} \left( s^K(\lambda) + \lambda K \sum_{i=0}^{K-1} e^K_{\lambda-1}(\lambda-i)\delta_i \right)
\]  

subject to:

\[
\sum_{i=0}^{K-1} \delta_i = 0.
\]  

One could reformulate this problem as a linear programming problem, however it is not usually necessary. Notice in the tables of Appendix C that the largest values of \( s^K(\lambda) \) occur for \( \lambda \) near the center of its range. The monotonicity theorem in section 7.3 of Chapter VII asserts that this is true in general. Therefore it should be expected that, if the \( \delta_i \) are small, the maximum in equation (9-25a) will occur for \( \lambda \) near its midpoint which is \( \frac{1}{2\lambda}(K-1) \). To minimize \( W_{EO} \) with respect to the variables \( \delta_i, i=0, 1, \ldots, K-1 \), it seems reasonable that the \( \delta_i \) be adjusted so that:

\[
W_{EO} = \sum_{i=0}^{K-1} \left( s^K(\lambda) + \lambda K \sum_{i=0}^{K-1} e^K_{\lambda-1}(\lambda-i)\delta_i \right)
\]  

for the \( K \) values of \( \lambda \) nearest the center of its range. What is
happening is that the K+1 variables $\delta_0, \delta_1, \ldots, \delta_{K-1}$ and $W_{EO}$ are adjusted, subject of course to constraint (9-25b), so that the K largest probabilities on the face of the $\lambda$-graph are all equal. Thus there are K+1 equations in K+1 unknowns. Unless the problem is quite singular, this is the best which can be done. In practice, the equations can be solved and then it can be verified that these were indeed the values of $\lambda$ at which the maximum in (9-25a) occurs.

This completes the analysis of the EO-Game. It has been shown that, when $\lambda=2$, the evader should choose his controls $u_1, u_2, \ldots, u_N$ to be mutually stochastically independent random variables uniformly distributed on $U$. The marksman should always choose the centerpoint of the face of the 2-graph as his aiming point:

$$\hat{z}_K = K-1.$$  

The Value $W_{EO}$ of the game is $1/K$.

When $\lambda > 2$ an approximate method for finding the solution, based on linearizing equation (9-16) about a near optimal point, was presented. It was shown that to find the Value of the game and the optimal evader's strategies requires the solution of a set of K+1 simultaneous linear equations in K+1 unknowns the coefficients of which are given in terms of the path function $g^K_k(\lambda)$.

9.5 Example

As an illustration, the equations are worked out here for the case $K = \lambda = 3$. From Table C-2 in Appendix C, the path function is:
The problem in equations (9-25) may be written:

\[ W_{EO} = \min_{\delta_1} \max_{\delta_1} S B(\delta) \]

subject to:

\[ \delta_0 + \delta_1 + \delta_2 = 0 \]

where:

\[
\begin{align*}
B(0) &= \frac{1}{27}[1 + 9(\delta_0)] \\
B(1) &= \frac{1}{27}[3 + 9(2\delta_0 + \delta_1)] \\
B(2) &= \frac{1}{27}[6 + 9(3\delta_0 + 2\delta_1 + \delta_2)] \\
B(3) &= \frac{1}{27}[7 + 9(2\delta_0 + 3\delta_1 + 2\delta_2)] \\
B(4) &= \frac{1}{27}[6 + 9(\delta_0 + 2\delta_1 + 3\delta_2)] \\
B(5) &= \frac{1}{27}[3 + 9(\delta_1 + 2\delta_2)] \\
B(6) &= \frac{1}{27}[1 + 9(\delta_2)].
\end{align*}
\]

It should be fairly clear from this formulation that the \( \delta_i \) are symmetric, i.e., \( \delta_0 = \delta_2 \), and that \( B(2), B(3), \) and \( B(4) \) are the \( K \) largest.
The equations corresponding to (9-26) are:

\[
27 \frac{W_{EO}}{S} - 27 \delta_0 - 18 \delta_1 - 9 \delta_2 = 6
\]

\[
27 \frac{W_{EO}}{S} - 18 \delta_0 - 27 \delta_1 - 18 \delta_2 = 7
\]

\[
27 \frac{W_{EO}}{S} - 9 \delta_0 - 18 \delta_1 - 27 \delta_2 = 6
\]

\[
\delta_0 + \delta_1 + \delta_2 = 0
\]

where \( \frac{W_{EO}}{S} \) is the average hit probability per shot and again it may be seen that \( \delta_0 = \delta_2 \). The solution is:

\[
\delta_0 = \delta_2 = \frac{1}{18} \quad \delta_1 = -\frac{1}{9}
\]

\[
\frac{W_{EO}}{S} = \frac{2}{9} = .2222
\]

so that the corresponding mixed strategy for the evader is:

\[
P_0 = p_2 = \frac{7}{18} = .3888
\]

\[
P_1 = \frac{2}{9} = .2222
\]

It should be remembered that this "solution" is only an approximate one, based on a linearized analysis. The exact solution found by numerical methods is:

\[
P_0 = p_2 = .374
\]

\[
P_1 = .252
\]

\[
\frac{W_{EO}}{S} = .2274
\]
As can be seen, the accuracy of the linearized analysis is surprisingly good. It can be expected that as $K$ and $\lambda$ increase the accuracy should improve rapidly since the probabilities of the nodes in the neighborhood of the midpoint of the face become more nearly equal and only very small values of $\delta_i$ are required to make them exactly equal. As $K$ and $\lambda$ increase $\delta_i$ decreases, which increases the accuracy quadratically since the first terms neglected in analysis were of the order of $\delta_i^2$.

In any case, an upper bound for $W_{EO}$ can be obtained by setting $\delta_0, \delta_1, \ldots, \delta_{K-1}$ in equation (9-25a) to zero, yielding:

$$W_{EO} \leq S \sum_{K}^{K} K^{-\lambda}.$$ 

If $K$ and $\lambda$ are large, this bound is fairly tight since the actual $\delta_i$ are small.

9.6 The El-Game

Although the analyses of the MO- and EO-Games were interesting, their solutions are not particularly good approximations to that of the original game. This can be seen by noting that the Value $W_{MO}$ of the MO-Game is the same as the lower bound derived in Chapter VIII while the Value $W_{EO}$ of the EO-Game is roughly the upper bound.

One further step in complexity are the finite memory games in which $R = 1$. In this section, the results of an analysis of the El-Game are described which, because its Value $W_{El}$ is somewhat less than $W_{EO}$, yield a lower and more satisfactory upper bound on $V$. 
In the El-Game, the evader is allowed to choose his control \( u_k \) at time \( k \) only as a function of its immediate predecessor \( u_{k-1} \). It cannot be a function of the other past controls \( u_1, u_2, \ldots, u_{k-2} \) because the evader has only one memory cell and must "forget" them in order to store \( u_{k-1} \). Since the evader will certainly use a mixed strategy, the control \( u_k \) will be a random variable. However, unlike the EO-Game, the probability density function of \( u_k \) may be conditioned upon the previous control \( u_{k-1} \) so that the sequence \( u_1, u_2, u_3, \ldots, u_N \) is a first order Markov Chain.

Suppose the transition probabilities chosen by the evader are:

\[
p_{ij} = \Pr(u_k = i \mid u_{k-1} = j) \quad i, j \in U.
\]

Naturally, they must satisfy the axioms of probability:

\[
\sum_{j=0}^{K-1} p_{ij} = 1 \quad j \in U \quad (9-27)
\]

\[
p_{ij} \geq 0 \quad i, j \in U.
\]

The steady-state probabilities of the \( u_k \) are defined:

\[
P_i = \Pr(u_k = i)
\]

and are the solutions of:

\[
P_i = \sum_{j=0}^{K-1} p_{ij} P_j \quad i = 0, 1, \ldots, K-1 \quad (9-28)
\]
Now, as was discussed in section 9.1, the marksman need only base his aiming point calculation on his $R$ most recent measurements and in this case $R = 1$ so that the marksman's problem is to choose the function $\psi_k(u_{k-1})$ which is his aiming point $\hat{z}_k$:

$$\hat{z}_k = \psi_k(u_{k-1}).$$

The Value of the El-Game is:

$$W_{1,1} = \min_{p_{ij}} \max_{\psi} \sum_{k=1}^{N} \Pr\{\text{hit at time } k\}$$

$$= \min_{p_{ij}} \sum_{k=1}^{N} \sum_{j=0}^{K-1} \left[ \max_{\psi} \Pr\{\hat{z}_k = z_j \mid u_{k-1} = j\} \right] p_{ij}$$

$$= \min_{p_{ij}} \sum_{k=1}^{N} \sum_{j=0}^{K-1} \left[ \max_{\psi} q_{k,j} \right] p_{ij}$$

where $q_{k,j}$ is the a posteriori probability density function:

$$q_{k,j} = \Pr\{z_k = \ell \mid u_{k-1} = j\}$$

$$= \Pr\{u_k + u_{k-1} + \ldots + u_{k-\lambda} = \ell \mid u_{k-1} = j\}$$

(9-29)

$$= \sum_{\ell} p_{i_1, i_2, \ldots, i_\lambda} \times \prod_{j=1}^{\lambda} p_{i_j}$$

and the summation is over all $i_1, i_2, \ldots, i_\lambda$ such that:

$$i_1 + i_2 + i_3 + \ldots + i_\lambda = \ell.$$

Suppose that $i^*_j$ is the value of $\ell$ at which $q_{k,j}$ takes on its maximum value:
\[
q_{k,j}^* = \max_k q_{k,j}
\]
then:
\[
q_k = \psi_k(u_{k-\lambda}) = \lambda^* u_{k-\lambda}
\]
and the Value is:
\[
W_{E_1} = \min_p \sum_{j=0}^{K-1} q_{k,j}^* p_j
\]
Solving this problem analytically is apparently very difficult. One is hampered by the very complex form of \(q_{k,j}\) in equation (9-29), the difficulty, inherent in all such eigenvalue problems, of finding \(p_j\) for arbitrary \(p_{ij}\) from (9-28), and the non-analytic form of equation (9-30).

After all attempts at solving the problem analytically proved futile, a numerical solution was obtained on the PDP-9 computer for a number of different combinations of \(K\) and \(\lambda\). This involved using a nonlinear programming algorithm to solve equation (9-31) subject to constraints (9-27). It proved not as straightforward as it sounds however because most nonlinear programming routines utilize the partial derivatives of the objective function or, at least require such derivatives to be continuous. However, equation (9-30) guarantees that such will not be the case.

In the end, the Simplex Method of Nelder and Mead [25] was employed and, since it makes no demands at all upon the function or its derivatives, it solved the problem nicely. Because such an algorithm has perhaps
some interest in its own right, the Fortran listing of it is included in Appendix D.

The program was run for a number of combinations of $K$ and $\lambda$. The resulting values of $W_{E1}$ are shown in Table 9-1. Not surprisingly, the Value and optimal strategies for the case $K = \lambda = 2$ were those of Isaacs' Game given in (3-21) and (3-22).

Table 9-1 $W_{E1}$ for various values of $K$ and $\lambda$.

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>.3820</td>
<td>.2963</td>
<td>.2380</td>
<td>.2056</td>
</tr>
<tr>
<td>3</td>
<td>.2354</td>
<td>.1680</td>
<td>.1346</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>.1690</td>
<td>.1189</td>
<td>.0943</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>.1298</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>.1065</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>.0919</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

According to equation (9-3), $W_{ER}$ is an upper bound on the Value of the original game. A pleasant result of the computer analysis was that, unlike $W_{E0}$ which was disappointing in this regard, $W_{E1}$ proved to be a substantially tighter upper bound than that found in Chapter VIII. This new "tighter" upper bound is shown, along with the original one, in Figure 9-2.

The conclusion is that the class of Markovian strategies for the evader is a fairly good one and, if the exact solution to the game is unavailable, a Markovian strategy is a reasonable approximation.
Figure 9-2  Tighter upper bound on the Value when $\lambda = 2$. 

Upper Bound from Figure 8-1

Tighter Upper Bound

Lower Bound from Figure 8-1
9.7 Conclusions

In this chapter, the concept of a finite memory game has been defined and its properties clarified. It has been shown that the usefulness of finite memory games arises from two of their properties: 1) if one player's memory of the past is restricted, then the other may voluntarily restrict his own memory as well without incurring any performance degradation; 2) the sequence formed by the Values of these games as the amount of memory increases, converges monotonically to V. The first of these properties facilitates the analysis of finite memory games while the second assures that such analysis is meaningful in terms of the original Austere Game.

There followed discussions of three finite memory games. First, the MO-Game was analyzed in depth and closed form expressions for its strategies and Value were presented. Then the EO-Game was studied by linearizing the hit probability about a point near the optimum and then solving the resulting linear equations for the optimum. Finally, the El game was analyzed numerically for a number of different values of K and λ and its Value calculated.

Of the three, only the Value of the El-Game proved sufficiently different from the bounds on V derived in Chapter VIII to warrant consideration of the El-Game as a possible approximation to the original Austere Game.
CHAPTER X

THE EXACT ANALYTICAL SOLUTION FOR K=\lambda=2

In this chapter, the exact analytical solution for all Austere Games in which K = \lambda = 2 is presented. It is derived by the method suggested in Chapter IX: A particular class of strategies is prescribed for the marksman, the maximum is found within this class, and the resulting payoff is calculated. This payoff is by equation (8-4) a lower bound on V. Then, in section 10.2, a particular strategy for the evader is suggested whose payoff is by equation (8-2) an upper bound on the Value of the game. That these two strategies are optimal is proven in 10.3 when it is shown that this upper bound on V is equal to the lower bound. The chapter concludes with the development of closed form expressions for the strategies and payoffs of both players and a discussion of the asymptotic form of these in games of long duration.

10.1 Optimal Strategy for the Marksman

In this section, a particular strategy for the marksman is presented. Later, in section 10.3 this strategy will be proven to be his optimal strategy. Now however, it will appear arbitrary and unjustified as it bears little relation to the strategies discussed in previous chapters. It was obtained by logical induction after a
detailed study of the linear programming solution given in Figure 6-3 and is described here so that the background necessary for the optimality proof will be available.

It has been pointed out several times that the marksman must, in general, calculate each aiming point \( z_k \) as a function of his whole past history of measurements. To aid him in this calculation, let the marksman utilize the three state sequential machine \( M \) whose input is his string of measurements. Let the three states of \( M \) be labeled a, b, and c. In addition let the particular state of \( M \) at time \( k \) be \( s_k \) and its input at that time be \( u_{k-2} \). Then \( M \) is completely defined by its state table shown in Figure 10-1.

\[
\begin{array}{ccc}
 & 0 & 1 \\
 a & b & c \\
b & a & c \\
c & b & a \\
\end{array}
\]

Figure 10-1 State table of the sequential machine \( M \).

A more graphical definition of \( M \) is by means of its state diagram given in Figure 10-2.
The initial state $s_2$ of the machine is state $a$.

Let the marksman choose his aiming point $\hat{z}_k$ as a function of the state $s_k$ of his machine at time $k$. Since he must use a mixed strategy, $\hat{z}_k$ must be a random variable whose probabilities are conditioned upon $s_k$. In particular, let the marksman's aiming rule be:

$$
\begin{align*}
\Pr(\hat{z}_k = 0 \mid s_k = a) &= r_k \\
\Pr(\hat{z}_k = 1 \mid s_k = a) &= 1 - 2r_k \\
\Pr(\hat{z}_k = 2 \mid s_k = a) &= r_k \\
\Pr(\hat{z}_k = 0 \mid s_k = b) &= 1/2 \\
\Pr(\hat{z}_k = 1 \mid s_k = b) &= 1/2 \\
\Pr(\hat{z}_k = 2 \mid s_k = b) &= 0 \\
\Pr(\hat{z}_k = 0 \mid s_k = c) &= 0 \\
\Pr(\hat{z}_k = 1 \mid s_k = c) &= 1/2 \\
\Pr(\hat{z}_k = 2 \mid s_k = c) &= 1/2
\end{align*}
$$

where the probability $r_k$ may be freely chosen by the marksman to maximize his payoff.
The minimum payoff which results from this strategy will now be calculated. Let the function \( F_m(s_m, u_{m-1}) \) be the minimum payoff in the game which begins at time \( m-1 \) in which the marksman's sequential machine starts in state \( s_m \) and the evader's initial control is \( u_{m-1} \):

\[
F_m(s_m, u_{m-1}) = \min_{u_m, u_{m+1}, \ldots, u_N} \sum_{k=m}^{N} \Pr\{\text{hit at time } k\}.
\]

Then, from the definition of the marksman's strategy, the functional equations:

\[
\begin{align*}
F_m(a, 0) &= \min\left[(1 - r_m)F_{m+1}(b, 0), (1 - 2r_m)F_{m+1}(b, 1)\right] \\
F_m(a, 1) &= \min\left[(1 - 2r_m)F_{m+1}(c, 0), (1 - r_m)F_{m+1}(c, 1)\right] \\
F_m(b, 0) &= \min\left[\left(\frac{1}{2} + F_{m+1}(a, 0)\right), \left(\frac{1}{2} + F_{m+1}(a, 1)\right)\right] \\
F_m(b, 1) &= \min\left[\left(\frac{1}{2} + F_{m+1}(c, 0)\right), \left(0 + F_{m+1}(c, 1)\right)\right] \\
F_m(c, 0) &= \min\left[\left(0 + F_{m+1}(b, 0)\right), \left(\frac{1}{2} + F_{m+1}(b, 1)\right)\right] \\
F_m(c, 1) &= \min\left[\left(\frac{1}{2} + F_{m+1}(a, 0)\right), \left(\frac{1}{2} + F_{m+1}(a, 1)\right)\right]
\end{align*}
\]

are obtained. The boundary condition is:

\[
F_{N+1} \equiv 0.
\]

Notice that, from equations (10-4c) and (10-4f):

\[
F_m(b, 0) = F_m(c, 1) \neq F_m^2
\]

so that, from equations (10-4d) and (10-4e):

\[
F_m(b, 1) = F_m(c, 0) \neq F_m^3
\]
and therefore, from equations (10-4a) and (10-4b):

\[ F_m(a,0) = F_m(a,1) \]

so that the equations become:

\[ F_m^1 = \min[(r_m + F_{m+1}^2), ((1-2r_m) + F_{m+1}^3)] \]  \hspace{1cm} (10-5a)
\[ F_m^2 = \frac{1}{2} + F_m^1 \]  \hspace{1cm} (10-5b)
\[ F_m^3 = \min[F_{m+1}^2, (\frac{1}{2} + F_{m+1}^3)] \]  \hspace{1cm} (10-5c)

Since the marksman may choose \( r_k \) to maximize the payoff, by Bellman's principle of optimality [22] he will choose it to maximize \( F_m^1 \) in (10-5a). Since \( F_m^1 \) is the minimum of two terms, one of which increases linearly with \( r_m \) while the other decreases linearly with \( r_m \), \( F_m^1 \) is maximized when the two terms are equal, yielding:

\[ r_m = \frac{1}{3}(1 - F_{m+1}^2 + F_{m+1}^3) \]  \hspace{1cm} (10-6)

so that the functional equations become:

\[ F_m^1 = \frac{1}{3} + \frac{2F_m^2}{3} + \frac{F_m^3}{3} \]  \hspace{1cm} (10-7a)
\[ F_m^2 = \frac{1}{2} + F_{m+1}^1 \]  \hspace{1cm} (10-7b)
\[ F_m^3 = \min[F_{m+1}^2, (\frac{1}{2} + F_{m+1}^3)] \]  \hspace{1cm} (10-7c)

Now, if it is assumed (subject to later confirmation) that

\[ F_m^2 < \frac{1}{2} + F_m^3 \] for all \( m \), then equations (10-7) reduce to:
a set of three, linear, first order difference equations. It is somewhat more convenient to write these difference equations using matrix notation:

\[ F_m = A F_{m-1} + \begin{bmatrix} 1/3 \\ 1/2 \\ 0 \end{bmatrix} F_{N+1} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (10-8) \]

where:

\[ F_m = \begin{bmatrix} F_1^m \\ F_2^m \\ F_3^m \end{bmatrix} \quad A = \begin{bmatrix} 0 & 2/3 & 1/3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \]

The value of the probability \( r_m \) is conveniently obtained via:

\[ r_m = \frac{1}{3} + [0 - \frac{1}{3} \frac{1}{3}] F_{m+1} \quad (10-9) \]

while the minimum payoff if the marksman uses this strategy is \( F_2^1 \) since the machine \( M \) starts in state \( a \) at time \( k = 2 \) yielding:

\[ W_M = [1 \ 0 \ 0] F_2. \quad (10-10) \]

This payoff is, by equation (8-4), a lower bound on the Value of the game:
\[ W_M \leq V. \]  

(10-11)

Later it will be shown that this always holds with equality.

To summarize: The marksman's optimal strategy (its optimality will be proven in section 10.3) is to build machine \( M \), defined in Figures 10-1 and 10-2, whose input is the string of his measurements. He should then choose his aiming point \( \hat{z}_k \) at time \( k \) as a random variable whose probabilities are conditioned upon \( s_k \) the internal state of \( M \) at time \( k \) in the manner given by equations (10-1), (10-2), and (10-3). To do this, he must solve the three dimensional, first order difference equation (10-8) backwards from the final condition at time \( N + 1 \). Once the solution \( F_m \) is known for \( m = 2, 3, 4, \ldots, N \) the aiming probability \( r_m \) may be calculated from equation (10-9) and the minimum payoff \( W_M \) (which will be shown to equal \( V \)) found from equation (10-10).

10.2 Optimal Strategy for the Evader

In this section, an apparently arbitrary strategy is proposed for the evader and the payoff \( W_E \) which results from its use is calculated. Later, it will be shown that he can do no better and therefore the strategy is optimal. As with the marksman's strategy in the last section, this also was obtained by induction on the linear programming solution of Figure 6-3 and therefore has little in common with the strategies presented in previous chapters. Its justification is simply that it is optimal.
The strategy is as follows. Let the evader generate the Markov Chain \( Y = (y_0, y_1, y_2, \ldots, y_N) \) in which \( y_i \) can have only three possible values or "states". Call these A, B, and C. Let the transition probabilities be:

\[
\begin{align*}
P_{AA} &= P_{BB} = P_{CC} = P_{CB} = P_{AC} = 0 \\
P_{AB} &= P_{BC} = 1 \\
P_{BA} &= 2/3 \\
P_{CA} &= 1/3
\end{align*}
\]

where:

\[
P_{ij} = \Pr\{y_k = i \mid y_{k-1} = j\}.
\]

These are more readily visualized by means of the state transition diagram in Figure 10-3.

![State transition diagram of Markov Chain Y](image)

Figure 10-3 State transition diagram of Markov Chain Y.

The initial state \( y_0 \) is B with probability one.

Once he has generated the Markov Chain \( Y \), let the evader choose his controls by means of the control law:

If \( y_k = A \): let \( u_k \) be a random variable with probabilities:

\[
\Pr\{u_k = 0\} = \Pr\{u_k = 1\} = 1/2
\]
If \( y_k = B \): let \( u_k = u_{k-1} \) \hspace{1cm} (10-13)

If \( y_k = C \): let \( u_k = u_{k-1} \)

where \( u_{k-1} = 1 - u_{k-1} \), the one's complement of \( u_{k-1} \).

An important piece of information about a Markov Chain is the probability \( P_k(i) \) of finding it in state \( i \) at time \( k \):

\[
P_k(i) = Pr(y_k = i) \quad i = A, B, C
\]

which may be calculated from:

\[
P_k = A^T P_{k-1} = (A^T)^k P_0
\]

where:

\[
P_k = \begin{bmatrix} P_k(A) \\ P_k(B) \\ P_k(C) \end{bmatrix} \quad P_0 = \begin{bmatrix} P_0(A) \\ P_0(B) \\ P_0(C) \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}
\]

and:

\[
A = \begin{bmatrix} 0 & 2/3 & 1/3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}
\]

Notice that the matrix \( A \) is the same as that which arose in the discussion of the marksman's strategy.

The maximum payoff \( W_E \) resulting from the use of this strategy will now be calculated. At time \( k \), the marksman knows the evader's controls \( u_1, u_2, u_3, ..., u_{k-2} \) and, therefore, a trivial computation allows him to infer the state \( y_{k-2} \) of the evader's Markov Chain.
Knowing the state $y_{k-2}$ and the evader's last control $u_{k-2}$ is all that he needs to determine his aiming point because this is all the evader has used to determine his future controls. In other words:

$$\hat{z}_k = \psi_k(u_{k-2}, y_{k-2}).$$

The payoff is:

$$W_E = \max_{\psi} \sum_{k=1}^{N} \Pr\{\text{hit at time } k\}$$

$$= \sum_{k=2}^{N} \sum_{j=0}^{1} \frac{1}{2} \sum_{i=A, B, C} \left[ \max_{\psi} \Pr\{z_k = z_k \mid y_{k-2} = i, u_{k-2} = j\} \right] p_{k-2}(i)$

$$= \sum_{k=2}^{N} \sum_{j=0}^{1} \frac{1}{2} \sum_{i=A, B, C} \left[ \max_{\psi} \Pr\{u_k + u_{k-1} = i \mid y_{k-2} = i, u_{k-2} = j\} \right] p_{k-2}(i)$$

where the hit probability $q_{ki j}$ is defined:

$$q_{ki j} = \Pr\{u_k + u_{k-1} = i \mid y_{k-2} = i, u_{k-2} = j\}$$

and advantage has been taken of the fact that, by symmetry, the two values of $u_{k-2}$ are equally likely.

The values of $q_{ki j}$ are most easily calculated with the aid of the 2-graphs shown in Figure 10-4. By symmetry, the values of $q_{ki j}$ may be found by simply reflecting these graphs about their horizontal centerlines.
$y_{k-2} = A, \ u_{k-2} = 0$:

$y_{k-2} = B, \ u_{k-2} = 0$:

$y_{k-2} = C, \ u_{k-2} = 0$:

Figure 10-4 Graphs used in calculating the hit probability $q_{xij}$. 
The information in these graphs can be succinctly summarized as:

If \( y_{k-2} = A \) or \( B \), the hit probability is 1/3 no matter what aiming point is chosen; however if \( y_{k-2} = C \), a hit probability of 1/2 may be expected. The resulting maximum payoff is:

\[
W_E = \sum_{k=2}^{N} \left( \frac{1}{3} P_{k-2}(A) + \frac{1}{3} P_{k-2}(B) + \frac{1}{2} P_{k-2}(C) \right)
\]

\[
= \sum_{k=2}^{N} \begin{bmatrix} 1/3 & 1/3 & 1/2 \end{bmatrix} P_{k-2}
\]

\[
= \sum_{k=0}^{N-2} \begin{bmatrix} 1/3 & 1/3 & 1/2 \end{bmatrix} (A^T)^k P_0.
\]

And finally since each term in this summation is a scalar and thus is equal to its transpose, \( W_E \) may be rewritten:

\[
W_E = P_0^T \sum_{k=0}^{N-2} A^k \begin{bmatrix} 1/3 \\ 1/3 \\ 1/2 \end{bmatrix}.
\]

As an aid to solving for \( W_E \), define the 3-vector \( E_m \) as:

\[
E_m = \sum_{k=0}^{m-1} A^k \begin{bmatrix} 1/3 \\ 1/3 \\ 1/2 \end{bmatrix}.
\]

Then, \( E_m \) may be found as a solution to the first order vector difference equation:

\[
E_m = A E_{m-1} + \begin{bmatrix} 1/3 \\ 1/3 \\ 1/2 \end{bmatrix}, \quad E_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{10-14}
\]

and the payoff is:
which, by equation (8-2), is an upper bound on the Value of the game:

\[ V \leq W_E \]  \hspace{1cm} (10-16)

It will be shown in the next section that this relation always holds with equality.

To summarize: The evader's optimal strategy (its optimality will be shown in 10.3) is to form the Markov Chain \( Y \) defined by its transition probabilities in equations (10-12) or its transition diagram in Figure 10-3. He should then choose his control \( u_k \) at time \( k \) by the control law given in (10-13). The maximum payoff if he uses this strategy will be \( W_E \) in equation (10-15) in which \( E_{N-1} \) can be found by solving the first order vector difference equation (10-14) forward from the initial condition at time 0.

10.3 Proof of Optimality

It is now possible to prove the optimality of these two strategies. This will be done by showing from an analysis of the difference equations (10-8) and (10-14) that \( W_E = W_M \).

It is well known that a 3-dimensional first order vector difference equation like (10-8) can be transformed into a third order scalar difference equation. If for example a difference equation in \( F^1_m \) the first component of \( F^m \) is desired, it may be obtained by performing the matrix multiplication and then repeatedly substituting:
so that the desired difference equation is:

\[ F_m^1 = \frac{2}{3} F_{m+2}^1 + \frac{1}{3} F_{m+3}^1 + \frac{1}{3} \]

\[ = \frac{2}{3} \left( F_{m+2}^{1} + \frac{1}{2} \right) + \frac{1}{3} \left( F_{m+2}^{2} \right) + \frac{1}{3} \]

\[ = \frac{2}{3} F_{m+2}^{1} + \frac{1}{3} \left( F_{m+3}^{1} + \frac{1}{2} \right) + \frac{2}{3} \]

whose boundary conditions are:

\[ F_{N-1}^1 = \frac{2}{3}, \quad F_N^1 = \frac{1}{3}, \quad F_{N+1}^1 = 0. \tag{10-18} \]

A similar technique can be used on the evader's 3-dimensional vector difference equation (10-14). Solving for \( E_m^2 \) the second component of \( E_m \) (the three components of \( E_m \) are \( E_m^1, E_m^2, \) and \( E_m^3 \) respectively) yields:

\[ E_m^2 = E_{m-1}^1 + \frac{1}{3} \]

\[ = \left( \frac{2}{3} E_{m-2}^2 + \frac{1}{3} E_{m-2}^3 + \frac{1}{3} \right) + \frac{1}{3} \]

\[ = \frac{2}{3} E_{m-2}^2 + \frac{1}{3} \left( E_{m-3}^2 + \frac{1}{2} \right) + \frac{2}{3} \]

so that the desired difference equation is:

\[ E_m^2 = \frac{2}{3} E_{m-2}^2 + \frac{1}{3} E_{m-3}^2 + \frac{5}{6} \tag{10-19} \]

with boundary conditions:
The similarity of difference equations (10-17) and (10-19) and their boundary conditions (10-18) and (10-20) means that:

\[ E_0^2 = 0, \quad E_1^2 = \frac{1}{3}, \quad E_2^2 = \frac{2}{3}. \]  

(10-20)

for all \( m \). In particular, it means that:

\[ E_0^2 = F_1, \quad E_{N-1}^2 = F_2 \]

which combined with (10-10) and (10-15) implies that:

\[ W_M = W_E. \]

But by equations (10-11) and (10-16):

\[ W_M \leq V \leq W_E \]

so that:

\[ W_M = W_E = V. \]  

(10-22)

Since the strategies prescribed above for the marksman and evader each yield the Value \( V \), they are both optimal which was to be shown.

10.4 Closed Form Solutions

The strategies given in sections 10.1 and 10.2 for the marksman and the evader are stated in terms of the solutions of difference equations (10-8) and (10-14), one for each participant. Although this
is a succinct and valuable format, it is often useful to have closed form expressions for the various quantities. In this section, the methods of classical difference equation analysis are applied to yield such closed form expressions.

Equation (10-8), the difference equation for the marksman, is clearly a linear first order vector difference equation whose forcing function (the vector \([1/3 \hspace{1em} 1/2 \hspace{1em} 0]^T\)) is a constant. The Total Solution to such a difference equation consists of the sum of a Particular Solution \(F_m^P\) due to the forcing function and a Homogeneous Solution \(F_m^H\) required to match the boundary conditions. This is written:

\[
F_m = F_m^P + F_m^H.
\]

In the case of equation (10-8) the Particular Solution is a ramp function:

\[
F_m^P = \begin{bmatrix}
5/14(N+1-m) \\
5/14(N+1-m) + 2/14 \\
5/14(N+1-m) - 3/14
\end{bmatrix}
\]

The Homogeneous Solution is:

\[
F_m^H = A^{N+1-m} F_{N+1}^H
\]

where \(F_{N+1}^H\) is a vector of arbitrary constants. If \(F_m\) is to match the boundary condition in (10-8), \(F_{N+1}^H\) must be:
and the Total Solution in closed form becomes:

\[ \begin{bmatrix} 5/14 (N+1-m) \\ 5/14 (N+1-m) \\ 5/14 (N+1-m) \end{bmatrix} + (I - A^{N+1-m}) \begin{bmatrix} 0 \\ 2/14 \\ -3/14 \end{bmatrix} \]  

(10-23)

where \( I \) is the 3 x 3 Identity Matrix. From this is obtained the aiming probability:

\[ r_m = \frac{3}{14} - [0 \ 1/3 \ +1/3] A^{N+1-m} \begin{bmatrix} 0 \\ 2/14 \\ -3/14 \end{bmatrix} \]  

(10-24)

and the minimum payoff:

\[ W_m = \frac{5}{14} (N-1) - [1 \ 0 \ 0] A^{N-1} \begin{bmatrix} 0 \\ 2/14 \\ -3/14 \end{bmatrix} \]  

(10-25)

A similar technique can be used for (10-14), the evader's difference equation. Its Particular Solution \( E^p_m \) is a ramp similar to that of the marksman:

\[ E^p_m = \begin{bmatrix} 5/14 m \\ 5/14 m - 1/42 \\ 5/14 m + 5/42 \end{bmatrix} \]

and its Homogeneous Solution is:

\[ E^H_m = A^m E^H_0 \]
where \( \mathbf{E}_0^H \) is a vector of arbitrary constants chosen to match the boundary condition in (10-14). If it is so chosen, then:

\[
\mathbf{E}_0^H = \begin{bmatrix}
0 \\
1/42 \\
-5/42
\end{bmatrix}
\]

and the Total Solution of the evader's difference equation is:

\[
\mathbf{E}_m = \begin{bmatrix}
5/14 m \\
5/14 m \\
5/14 m
\end{bmatrix} + (\mathbf{I} - \mathbf{A}^m) \begin{bmatrix}
0 \\
-1/42 \\
5/42
\end{bmatrix}
\]

The resulting maximum payoff from (10-15) is:

\[
W_E = \frac{5}{14}(N-1) - \frac{1}{42} - [0 \ 1 \ 0] \mathbf{A}^{N-1} \begin{bmatrix}
0 \\
-1/42 \\
5/42
\end{bmatrix}
\]

(10-27)

In this section, closed form expressions were derived for \( \mathbf{E}_m \) and \( \mathbf{F}_m \), the solutions of the evader's and marksman's difference equations. Each was obtained using a classical difference equation technique: first the particular solution corresponding to the equation's forcing function was found; then the homogeneous solution was derived; finally the two were added together and the arbitrary constants adjusted to match the boundary conditions.

Closed form expressions for the other parameters which depend upon \( \mathbf{E}_m \) and \( \mathbf{F}_m \) were then obtained. These included \( r_m \), \( W_M \), and \( W_E \). Since the latter two are equal to \( V \), the Value of the game, they are especially satisfying to have in closed form.
10.5 Asymptotic Form in Games of Long Duration

In this section, the asymptotic form of the Game, its Value and the optimal strategies, as \( N \to \infty \) is obtained. It is derived from and made possible by the closed form solutions which were found in the previous section.

A study of equations (10-23), (10-24), (10-25), (10-26), and (10-27) reveals that the central difficulty in obtaining asymptotic expressions for them is the problem of evaluating expressions whose form is:

\[
y = A^m x
\]

when \( m \) is very large. Fortunately, the results of linear algebra are available to take the edge off such difficulties.

If the eigenvalues \( \lambda_1, \lambda_2 \) and \( \lambda_3 \) of \( A \) are distinct, then the corresponding eigenvectors \( e_1, e_2 \) and \( e_3 \) are unique and linearly independent so that they span the vector space and may be used as a basis. Suppose that the vector \( x \) is expressed in terms of this basis as:

\[
x = c_1 e_1 + c_2 e_2 + c_3 e_3.
\]

Then the vector \( y \) can be written:

\[
y = A^m x = c_1 \lambda_1^m e_1 + c_2 \lambda_2^m e_2 + c_3 \lambda_3^m e_3.
\]
Now, if the eigenvalues are ordered so that:

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3|$$

then, if \( m \) is large enough, the first term in (10-29) is the predominant one and:

$$y \to c_1 \lambda_1^m e_1. \quad (10-30)$$

To apply this result, the eigenvalues and eigenvectors of \( A \) must be found. It is easily verified that these are:

$$\begin{align*}
\lambda_1 &= 1 \\
\lambda_2 &= -\frac{1}{2} + j\sqrt{\frac{1}{12}} \\
\lambda_3 &= -\frac{1}{2} - j\sqrt{\frac{1}{12}}
\end{align*}$$

$$
\begin{bmatrix}
1 \\
1 \\
1
\end{bmatrix} \quad \begin{bmatrix}
\frac{1}{6} - j\sqrt{\frac{1}{12}} \\
-\frac{1}{2} + j\sqrt{\frac{1}{12}} \\
1
\end{bmatrix} \quad \begin{bmatrix}
\frac{1}{6} + j\sqrt{\frac{1}{12}} \\
-\frac{1}{2} - j\sqrt{\frac{1}{12}} \\
1
\end{bmatrix}
$$

where \( j = \sqrt{-1} \). Clearly, the requirements on the eigenvalues are satisfied: they are distinct and ordered so that \( \lambda_1 \) is the largest in magnitude (the magnitude of both \( \lambda_2 \) and \( \lambda_3 \) is \( \sqrt{1/3} \)). The next step is to find \( c_1, c_2 \) and \( c_3 \) in (10-28). However, note that since the asymptotic form to be used is (10-30), only \( c_1 \) is necessary —— a fact which allows a considerable saving in computation.

Consider the vector \( d \):

$$
\begin{bmatrix}
3/7 \\
3/7 \\
1/7
\end{bmatrix}
$$
This vector has the property that:

\[ d^T e_1 = 1 \quad d^T e_2 = 0 \quad d^T e_3 = 0 \]

and hence it is the **dual vector** to eigenvector \( e_1 \). Applying it to (10-28):

\[ d^T x = d^T (c_1 e_1 + c_2 e_2 + c_3 e_3) = c_1 \]

suggests an easy way of computing \( c_1 \). Combining all these facts yields the limiting form:

\[ y = A^m x \rightarrow d^T x \lambda^m_1 e_1 = d^T x \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \]

which is valid for very large \( m \).

This fundamental formula can be used to calculate asymptotic forms for all the quantities which have closed form solutions given in section 10.4. In particular, for large \( m \) and larger \( N \):

\[
\begin{align*}
F_m & \rightarrow \begin{bmatrix}
\frac{5}{14} (N+1-m) - \frac{3}{98} \\
\frac{5}{14} (N+1-m) + \frac{11}{98} \\
\frac{5}{14} (N+1-m) - \frac{24}{98}
\end{bmatrix} \\
E_m & \rightarrow \begin{bmatrix}
\frac{5}{14} m - \frac{2}{294} \\
\frac{5}{14} m - \frac{9}{294} \\
\frac{5}{14} m + \frac{33}{294}
\end{bmatrix} \\
V & \rightarrow \frac{5}{14} S - \frac{3}{98} \\
\tau_m & \rightarrow \frac{3}{14}
\end{align*}
\]
where the total number of shots $S = N - \lambda + 1$ is in this case simply $N - 1$.

These are quite important results. Equation (10-32) says that, in games of long duration, the average hit probability per shot $V/S$ is:

$$V/S \rightarrow \frac{5}{14} = .3571.$$  

This compares with:

- Upper Bound from Chapter VIII = .5000
- Upper Bound from Chapter IX = .3820
- Solution to Isaacs' Game = .3820
- Lower Bound from Chapter VIII = .3333

Thus it can be seen that the actual solution lies almost exactly midway between the best upper and lower bounds presently known. It would not be too surprising if this would prove true for games with other values of $K$ and $\lambda$ when they are eventually solved.

10.6 Example

As an example, consider the game in which $K = \lambda = 2$ and $N = 4$. The marksman is allowed $S = 3$ shots and each aiming point must be chosen from among $L = 3$ possibilities. The solutions to the difference equations (10-8) and (10-14) are:
\[ \begin{bmatrix} E_0 & 0 \\ 0 & 0 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} E_1 & 1/3 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} E_2 & 13/18 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} E_3 & 19/18 \end{bmatrix} \begin{bmatrix} F_5 & 0 \\ 0 & 1/3 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} F_4 & 1/2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} F_3 & 4/6 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} F_2 & 19/18 \end{bmatrix} \begin{bmatrix} F_1 & 1/3 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} F_0 & 19/18 \end{bmatrix} \]

from which is obtained:

\[ W_M = [1 \ 0 \ 0] \begin{bmatrix} F_2 & 19/18 \end{bmatrix} \]

\[ W_E = [0 \ 1 \ 0] \begin{bmatrix} E_3 & 19/18 \end{bmatrix} . \]

As it should be, \( W_M \) is equal to \( W_E \) so that the Value of the game is:

\[ V = 19/18 . \]

The aiming probability \( r_k \) which the marksman should use whenever his sequential machine \( M \) is in state \( a \) is found from (10-9) to be:

\[ r_2 = 2/9 \quad r_3 = 1/6 \quad r_4 = 1/3 . \]

The resulting aiming strategy for the marksman is given in Table 10-1.

<table>
<thead>
<tr>
<th>( k )</th>
<th>marksman's measurements</th>
<th>( s_k )</th>
<th>probability of ( \hat{z}_k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>--- --- ---</td>
<td>a</td>
<td>2/9 5/9 2/9</td>
</tr>
<tr>
<td>3</td>
<td>( u_1 = 0 )</td>
<td>b</td>
<td>1/2 1/2 0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>c</td>
<td>0 1/2 1/2</td>
</tr>
<tr>
<td>4</td>
<td>( u_1u_2 = 00 )</td>
<td>a</td>
<td>1/3 1/3 1/3</td>
</tr>
<tr>
<td>4</td>
<td>( 01 )</td>
<td>c</td>
<td>0 1/2 1/2</td>
</tr>
<tr>
<td>4</td>
<td>( 10 )</td>
<td>b</td>
<td>1/2 1/2 0</td>
</tr>
<tr>
<td>4</td>
<td>( 11 )</td>
<td>a</td>
<td>1/3 1/3 1/3</td>
</tr>
</tbody>
</table>
Assuming that the marksman uses this aiming strategy the payoff for the various choices by the evader of the binary 4-tuple \( u = (u_1 u_2 u_3 u_4) \) is shown in Table 10-2.

<table>
<thead>
<tr>
<th>( u_1 u_2 )</th>
<th>( u_3 u_4 )</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>19/18</td>
<td>19/18</td>
<td>19/18</td>
<td>19/18</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>19/18</td>
<td>28/18</td>
<td>19/18</td>
<td>19/18</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>19/18</td>
<td>19/18</td>
<td>28/18</td>
<td>19/18</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>19/18</td>
<td>19/18</td>
<td>19/18</td>
<td>19/18</td>
<td></td>
</tr>
</tbody>
</table>

Naturally, the evader should select \( u \) to minimize the payoff. However, Table 10-2 shows that 19/18 is the lowest payoff he can obtain.

The Markov Chain \( Y \) combined with the control law promulgated in section 10.2 yields the optimal mixed strategy for the evader shown in Table 10-3.

<table>
<thead>
<tr>
<th>( u_1 u_2 u_3 u_4 )</th>
<th>( y_1 y_2 y_3 y_4 )</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0</td>
<td>A B A B</td>
<td>4/36</td>
</tr>
<tr>
<td>0 0 0 1</td>
<td>A B A C</td>
<td>2/36</td>
</tr>
<tr>
<td>0 0 1 0</td>
<td>A B A C</td>
<td>2/36</td>
</tr>
<tr>
<td>0 0 1 1</td>
<td>A B A B</td>
<td>4/36</td>
</tr>
<tr>
<td>0 1 0 0</td>
<td>A C - -</td>
<td>0</td>
</tr>
<tr>
<td>0 1 0 1</td>
<td>A C - -</td>
<td>0</td>
</tr>
<tr>
<td>0 1 1 0</td>
<td>A C B A</td>
<td>1/12</td>
</tr>
<tr>
<td>0 1 1 1</td>
<td>A C B A</td>
<td>1/12</td>
</tr>
<tr>
<td>1 0 0 0</td>
<td>A C B A</td>
<td>1/12</td>
</tr>
<tr>
<td>1 0 0 1</td>
<td>A C B A</td>
<td>1/12</td>
</tr>
<tr>
<td>1 0 1 0</td>
<td>A C - -</td>
<td>0</td>
</tr>
<tr>
<td>1 0 1 1</td>
<td>A C - -</td>
<td>0</td>
</tr>
<tr>
<td>1 1 0 0</td>
<td>A B A B</td>
<td>4/36</td>
</tr>
<tr>
<td>1 1 0 1</td>
<td>A B A C</td>
<td>2/36</td>
</tr>
<tr>
<td>1 1 1 0</td>
<td>A B A C</td>
<td>2/36</td>
</tr>
<tr>
<td>1 1 1 1</td>
<td>A B A B</td>
<td>4/36</td>
</tr>
</tbody>
</table>
Assuming that the evader uses this mixed strategy, the payoff for the various possible aiming points of the marksman may be calculated. These are shown in Table 10-4.

Table 10-4 Hit probabilities as a function of marksman's aiming point.

<table>
<thead>
<tr>
<th>k</th>
<th>measurements</th>
<th>$z_k$</th>
<th>$\hat{z}_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>-- -- --</td>
<td>1/3</td>
<td>1/3</td>
</tr>
<tr>
<td>3</td>
<td>$u_1 = 0$</td>
<td>1/3</td>
<td>1/3</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1/3</td>
<td>1/3</td>
</tr>
<tr>
<td>4</td>
<td>$u_1 u_2 = 01$</td>
<td>1/3</td>
<td>1/3</td>
</tr>
<tr>
<td></td>
<td>00</td>
<td>1/3</td>
<td>1/3</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>0</td>
<td>1/2</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>1/2</td>
<td>1/2</td>
</tr>
</tbody>
</table>

Naturally, the marksman should select his aiming points to maximize the payoff which is the expected number of hits. However, a short calculation shows that $19/18$ is the highest payoff attainable.

The reader might find it instructive to compare these results with the linear programming solution of the same game given in Figure 6-3. Notice that the probabilities of the marksman's aiming points in Table 10-1 exactly correspond with the optimal $p$ vector in Figure 6-3. However, the probabilities of the various controls in Table 10-2 are not the same as the optimal $p$ vector in Figure 6-3. There is nothing surprising about this. It has never been claimed that the evader's optimal mixed strategy is unique. The $p$ vector
in Figure 6-3 and the strategy in Table 10-3 are simply two different optimal solutions.

10.7 Summary

In this chapter, perhaps the most important chapter of the dissertation, the exact analytical solution of the Austere Game in which K and λ are both two has been described and discussed. The marksman's optimal policy, it was shown, is to utilize a three state sequential machine $M$ whose inputs are his measurements. His aiming point $z_k$ is then chosen randomly with probabilities determined by the state of $M$ at time $k$. The evader, on the other hand, should first generate a Markov Chain $Y$ with the particular transition probabilities specified by equation (10-12). He then can obtain his optimal control $u_k$ as a function of $y_k$, the state of $Y$ at time $k$.

This is a very satisfying solution by virtue of its elegant simplicity and the ingenious way in which state of $M$ and therefore the marksman's aiming points are effected by the whole past trajectory. The author firmly believes that Austere Games with other values of $K$ and $\lambda$ than two, when they are eventually solved, will exhibit similar structure.
CHAPTER XI
SUMMARY AND CONCLUSIONS

In these pages a version of the problem of aiming and evasion has been defined and discussed and a number of new results on it have been obtained. The problem of aiming and evasion involves a conflict between two adversaries: a marksman and an evader. The stationary marksman desires to predict the future position of the evader on the basis of present and past data. The mobile evader, on the other hand, must dodge and weave continuously for he would like the marksman’s predictions to be as inaccurate as possible. These two desiderata are clearly conflicting — the marksman and the evader cannot both have their way — and thus the problem is under the aegis of game theory.

The mathematical definition of the problem took place in several stages. In the first, a very general formulation called the Proto-problem was given. This was characterized by a discrete integer-valued time variable, fixed prespecified initial and final times, arbitrary state equations, a finite set of feasible controls, an additive payoff, and a saddle-point definition of optimality. Following this, the simplification obtained when the state equations are linear and the payoff is origin-independent was described. A particular version of this class called the Austere Game was then presented. This involves a particularly simple scalar state equation and a payoff defined to be the total number of direct hits.

166
Although the Austere Game was the focus of the dissertation, it is important to understand the whole sequence of aiming and evasion games, from the most general to the most specific, so that the role of each simplifying assumption is clarified and the objective, a general solution of all aiming and evasion games, may be kept in mind. The advantage of this type of presentation is that it shows what problems should be attacked next after the Austere Game is completely solved. Thus, in a very real sense, the sequence of games outlined here represents a whole program of research which starts with the last and simplest problem and grapples with successively more and more complex versions until the whole sequence is solved. This dissertation, a study of Austere Games, represents the first step in such a program. Moreover, the delineation of the program is in itself a contribution since it apparently has not been done before.

After the problem was defined, the existence of a solution was proven. This followed from the finiteness of the game so that von Neumann's fundamental theorem of two-person zero-sum game theory was applicable. In the process however, it was discovered that the normalized form for even the smallest games is inconceivably large so that it is useless as a tool for solving games.

At this point a new and novel notational device, the "sub-normalized" form was presented. Midway between the "normalized" and the "extensive" forms, the sub-normalized form is many orders of magnitude smaller than the former while being more useful than the latter. Its advantages are
threefold: notational, numerical, and conceptual. Its notational advantage is that it offers a way of writing the characteristics of the game in a fairly standardized way. In addition, it was shown that most of the numerical techniques designed to work on games in normalized form work on games in sub-normalized form as well so that, since the sub-normalized form for these games is a manageable size, the numerical techniques may be utilized. Finally, it was shown that much of the structure of the Austere Game is illuminated by the sub-normalized form, making it a useful conceptual tool.

Two computational algorithms were given for solving Austere Games. The first, linear programming, can be applied to any game in sub-normalized form. This involves finding the evader's optimal strategy as the solution of a linear program and the marksman's optimal strategy as the solution of a dual linear program. This furnishes a useful way to solve aiming and evasion games since efficient numerical techniques exist for solving linear programming problems.

Unfortunately, it was discovered that the size of the LP problem grows exponentially with N, the duration of the game so that linear programming apparently cannot be used successfully for games of long duration.

To circumvent this problem, a second technique, dynamic programming, was presented. The computational effort here grows only linearly with N since the algorithm requires one to solve a sequence of \( S = N - \lambda + 1 \).
problems but the size of each is fixed. This advantage is paid for however, by the fact that no such efficient numerical techniques exist for dynamic programming similar to those for linear programming. In addition, the size of the dynamic programming problem grows exponentially with the time delay \( \lambda \) so that the "curse of dimensionality" appears quickly in games with large \( \lambda \).

Next, graph theory was used to study the structure of the Austere Game. From this a strong symmetry result was proven. It was also found that the number of paths through the graph of a game was an important parameter and the "path function" which measures this number was defined and studied.

Both of these results had immediate applications. In Chapter VIII, bounds on \( V \), the Value of the game, were obtained. These were expressed in terms of the path function, and are quite useful since from them one can obtain valuable insight about the magnitude of \( V \) and its dependence on the parameters of the game. It was shown that in games of long duration, \( V \) is approximately proportional to \( N \) and that it decreases monotonically in an apparently complicated way as the other parameters \( K \) and \( \lambda \) are increased.

One of the reasons for the great technical difficulty of aiming and evasion games is that each marksman's aiming point and each evader's control must apparently depend on the whole past history of the evader's trajectory. How can this be done? The attempts to approximate it by having the strategies depend on only a finite amount of the most recent
past are documented in Chapter IX although they met with indifferent success. It was not until the exact analytical solution for the case where $K = \lambda = 2$ was obtained that a glimmering of the answer began to appear. It was found that the marksman should employ a finite state sequential machine whose input is the string of his measurements. The internal state of this machine apparently contains all necessary information about the past trajectory, for the marksman's aiming points depend solely upon it. Thus, the machine's internal state is a sufficient statistic for the whole past trajectory.

The solution in the case of the evader is similar. The evader should generate a Markov Chain with specified transition probabilities and then choose his individual controls as functions of the state of this Markov Chain. The state of the Markov Chain apparently contains all necessary information about the past trajectory and thus for the evader it is a sufficient statistic.

It is exciting to consider the question of whether the exact solution for other $K$ and $\lambda$ will have this same structure for if so, it represents an elegant blend of game theory, control theory, and automata theory.

This then is the author's sole suggestion for further research: an investigation of the Austere Game for other values of $K$ and $\lambda$ to see if the structure of the solution remains the same. Unfortunately, he has few suggestions as to how this may be done.
The reason is that the solution for \( K = \lambda = 2 \) was not derived. It was obtained by induction after a long study of the linear programming solution of the game in which \( K = \lambda = 2 \) and \( N = 4 \). There appear to be two possibilities: One is to solve by linear programming some typical games with other \( K \) and \( \lambda \) and hope to be able to see the solution by induction, perhaps with the aid of the insight already developed. The other is to make a close careful study of the solution for \( K = \lambda = 2 \) with the object of obtaining it by deduction rather than induction. Hopefully then, one could utilize the same deductive argument for other \( K \) and \( \lambda \) as well. In reality, neither technique seems highly promising.

Even if a general solution of the Austere Game were obtained, the other games outlined in Chapter III remain to be investigated. Thus, although this study represents a large step forward in the understanding of aiming and evasion games, much work remains to be done.
APPENDIX A

PROOF OF THE LINEAR SEPARABILITY OF

\[ F_k(s_{k-1}) \] IN EQUATION (6-23)

In this appendix, equation (6-23) of the text is proven. This equation states that the function \( F_k(s_{k-1}) \) is linearly separable:

\[
F_k(s_{k-1}) = \sum_{\ell=0}^{K^{k-1}} G_k(s_{k-1} \eta, s_{k-1} \eta+1, \ldots, s_{k-1} (\eta+K^{k-1}-1))
\]

where \( \eta = \ell K^{k-1} \).

The proof is by induction on \( k \). The theorem is certainly true for \( F_{N+1}(s_N) \) by equation (6-22e). Assume that it is true for \( k=m+1 \). Then equations (6-22) become:

\[
F_m(s_{m-1}) = \min_{s_m} \left[ \sum_{\beta} q_{m\beta} + \sum_{\ell=0}^{K^{m-1}-1} G_{m+1}(s_m \eta, s_m \eta+1, \ldots, s_m (\eta+K^{m-1}-1)) \right]
\]

subject to:

\[
\sum_{\xi} b_{m\beta \gamma} s_m \xi q_{m\beta} \leq 0
\]

\[
\sum_{\ell=0}^{K-1} s_m \xi K \xi = s_{m-1} \xi
\]

\[
s_m \xi > 0
\]
where the summation over B goes from 0 to \(K^{m-\lambda}-1\) while the one over \(\xi\) goes from 0 to \(K^m-1\). Both the objective function above and the constraints may be rewritten to yield:

\[
F_{m}(s_{m-1}) = \min_{s_{m} q_{mB}} \left[ q_{mB} + \sum_{\xi=\beta K}^{(\beta+1)\lambda-1} G_{m+1}(s_{m \eta}, \ldots, s_{m (\eta+K^{\lambda-1} - 1)}) \right]
\]

subject to:

\[
\sum_{\xi=\zeta}^{\beta K^{\lambda}-1} b_{mB} \gamma_{\xi} s_{m\xi} - q_{mB} \leq 0
\]

\[
\sum_{i=0}^{K-1} s_{m (\xi K+i)} = s_{m-1} \xi
\]

\[
s_{m\xi} \geq 0
\]

where \(\zeta = \beta K^{\lambda}\) and advantage has been taken of the fact that \(b_{mB} \gamma_{\xi}\) is zero outside the range:

\[
\beta K^{\lambda} \leq \xi \leq (\beta+1)K^{\lambda}-1.
\]

If it is noticed that \(q_{mB}\) is, by equation (A-2), only a function of \(s_{m (\zeta+j)}\) \(j = 0, 1, \ldots, K^{\lambda}-1\) then the equations can be rewritten:

\[
F_{m}(s_{m-1}) = \sum_{\beta} \min_{s_{m} q_{mB}}, s_{m\zeta}, \ldots, s_{m (\zeta+K^{\lambda}-1)}
\]

\[
+ \sum_{i=0}^{K-1} G_{m+1}(s_{m (\zeta+i K^{\lambda}-1) \ldots, s_{m (\zeta+i K^{\lambda}-1) + 1)}, \ldots, s_{m (\zeta+i K^{\lambda}-1) + K^{\lambda}-1})
\]

subject to:
Now if \( G_m \) is defined as the solution of the dynamic programming problem:

\[
G_m(s_{m-1}, s_{m-1}, \ldots, s_{m-1}, (K^\lambda-1)_{-1})
\]

\[
= \min_{s_m 0', s_m 1', \ldots, s_m (K^\lambda-1), q_m 0}
\]

\[
+ \sum_{i=0}^{K-1} G_{m+1}(s_m (iK^\lambda-1), s_m (iK^\lambda-1+1), \ldots, s_m (iK^\lambda-1+k-1))
\]

subject to:

\[
\sum_{j=0}^{K^\lambda-1} b_m \gamma_j s_{mj} - q_m 0 \leq 0
\]

\[
\sum_{i=0}^{K-1} s_m (iK+1) = s_{m-1} j
\]

\[
\geq 0
\]

and if it is noticed that, by its definition:
\[ b_{m+\gamma} \equiv b_{m+\gamma} \]

then:

\[ F_m(s_{m-1}) = \sum_{\beta} G_m(s_{m-1} \theta, s_{m-1} \theta+1, \ldots, s_{m-1} (\theta+K^{\lambda-1}-1)) \]

where \( \theta = 8K^{\lambda-1} \) and by the induction hypothesis equation (A-1) is proven.
APPENDIX B

DERIVATION OF THE CLOSED FORM OF

THE PATH FUNCTION

In this appendix, the closed form of the path function $g^K_k(\ell)$ in equation (7-2) is derived from its word statement definition:

$g^K_k(\ell)$ is the number of ways of distributing

$\ell$ non-distinct objects in $k$ cells such that

no cell contains more than $K-1$ objects.

The derivation utilizes a well known combinatorial result known as "the principle of inclusion and exclusion" [23].

Let: $N$ be the number of ways of distributing $\ell$ objects among $k$ cells.

Let: $a^i_1$ be the property that cell $i$ contains more than $K-1$ objects.

Let: $a^i_1'$ be the property that cell $i$ does not contain more than $K-1$ objects.

Let: $N(p_1, q_2 ... r)$ be the number of distributions having properties

$p_1, q_2, ..., r$ simultaneously.

In terms of these definitions, the path function is:

$g^K_k(\ell) = N(a'_1, a'_2, a'_3 ... a'_k)$

and, by the principle of inclusion and exclusion:

$g^K_k(\ell) = N(a'_1, a'_2, a'_3 ... a'_k)$

$= N - N(a'_1) - N(a'_2) - ... - N(a'_k)$

$+ N(a'_1 a'_2) + N(a'_1 a'_3) + ... + N(a'_1 a'_k)$

$+ N(a'_2 a'_3) + N(a'_2 a'_4) + ... + N(a'_2 a'_k)$
\[ + \ldots + \ldots + \ldots + N(a_{k-1} \ldots a_k) \]
\[ - N(a_1 a_2 a_3) - \ldots - N(a_{k-2} a_{k-1} a_k) \]
\[ + \ldots + \ldots + \ldots + \]
\[ + (-1)^k N(a_1 a_2 \ldots a_k) \]

Now, since the cells are all the same:

\[ N(a_i) = N(a_j) \]
\[ N(a_i a_j) = N(a_m a_n) \]
\[ \vdots \]

and the path function may be written:

\[ g_k^L = N - \binom{k}{1} N(a_1) + \binom{k}{2} N(a_1 a_2) - \binom{k}{3} N(a_1 a_2 a_3) \]
\[ + \ldots + (-1)^k \binom{k}{k} N(a_1 a_2 \ldots a_k) \]

where \( \binom{k}{m} \) is the number of different combinations of \( k \) objects taken \( m \) at a time:

\[ \binom{k}{m} = \frac{k!}{m! (k-m)!} \]

Now it is easily shown that:

\[ N = \binom{k+\ell-1}{\ell} \]

\[ N(a_1) = \begin{cases} 
\binom{k+(\ell-K)-1}{(\ell-K)} & \text{if } \ell \geq K \\
0 & \text{if } \ell < K 
\end{cases} \]

and, in general:
\[ N(a_1 a_2 \ldots a_m) = \begin{cases} 
\binom{k+(\ell-mK)-1}{\ell-mK} & \text{if } \ell \geq mK \\
0 & \text{if } \ell < mK .
\end{cases} \]

When these are substituted into the equation for \( g_k(\ell) \) above, equation (7-2) results and the derivation is complete.
APPENDIX C
TABLES OF THE PATH FUNCTION

The following tables give the values of the path function $g^K_k(l)$ for various values of the parameters $K, k,$ and $l$.

Table C-1 The path function $g^K_k(l)$ for $K = 2$.

<table>
<thead>
<tr>
<th>$l$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
<td>84</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>15</td>
<td>35</td>
<td>70</td>
<td>126</td>
<td>126</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>6</td>
<td>21</td>
<td>56</td>
<td>126</td>
<td>126</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>7</td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

179
Table C-2 The path function $g_k^n$ for $K = 3$.

<table>
<thead>
<tr>
<th>$l$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>36</td>
<td>45</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>7</td>
<td>16</td>
<td>30</td>
<td>50</td>
<td>77</td>
<td>112</td>
<td>156</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>6</td>
<td>19</td>
<td>45</td>
<td>90</td>
<td>161</td>
<td>266</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>16</td>
<td>51</td>
<td>126</td>
<td>266</td>
<td>504</td>
<td>832</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>10</td>
<td>45</td>
<td>131</td>
<td>357</td>
<td>784</td>
<td>1554</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>30</td>
<td>126</td>
<td>393</td>
<td>1016</td>
<td>2304</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>15</td>
<td>90</td>
<td>157</td>
<td>1107</td>
<td>2907</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>50</td>
<td>266</td>
<td>1616</td>
<td>3139</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>21</td>
<td>161</td>
<td>734</td>
<td>2907</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>6</td>
<td>77</td>
<td>504</td>
<td>1554</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>28</td>
<td>266</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>7</td>
<td>112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table C-3 The path function $g^k_\ell(\lambda)$ for $K = 4$.

<table>
<thead>
<tr>
<th>$\ell$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>36</td>
<td>45</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
<td>84</td>
<td>120</td>
<td>165</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>12</td>
<td>31</td>
<td>65</td>
<td>120</td>
<td>203</td>
<td>322</td>
<td>486</td>
<td>1206</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>12</td>
<td>40</td>
<td>101</td>
<td>216</td>
<td>413</td>
<td>728</td>
<td>1206</td>
<td>2593</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>10</td>
<td>44</td>
<td>135</td>
<td>336</td>
<td>728</td>
<td>1426</td>
<td>2593</td>
<td>5286</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>40</td>
<td>155</td>
<td>456</td>
<td>1128</td>
<td>2472</td>
<td>4950</td>
<td>9820</td>
<td>19660</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>31</td>
<td>155</td>
<td>516</td>
<td>1554</td>
<td>3823</td>
<td>8151</td>
<td>16301</td>
<td>32602</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>20</td>
<td>135</td>
<td>580</td>
<td>1915</td>
<td>5328</td>
<td>13051</td>
<td>30267</td>
<td>60534</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>101</td>
<td>516</td>
<td>2128</td>
<td>6728</td>
<td>18351</td>
<td>36702</td>
<td>73404</td>
<td>146808</td>
</tr>
<tr>
<td>11</td>
<td>4</td>
<td>65</td>
<td>456</td>
<td>2128</td>
<td>7728</td>
<td>23607</td>
<td>47214</td>
<td>94428</td>
<td>188856</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>35</td>
<td>336</td>
<td>1918</td>
<td>6092</td>
<td>12184</td>
<td>24368</td>
<td>48736</td>
<td>97472</td>
</tr>
<tr>
<td>13</td>
<td>15</td>
<td>216</td>
<td>1554</td>
<td>7728</td>
<td>30276</td>
<td>60552</td>
<td>121104</td>
<td>242208</td>
<td>484416</td>
</tr>
<tr>
<td>14</td>
<td>5</td>
<td>120</td>
<td>1128</td>
<td>6728</td>
<td>30276</td>
<td>60552</td>
<td>121104</td>
<td>242208</td>
<td>484416</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>56</td>
<td>728</td>
<td>5328</td>
<td>27876</td>
<td>55753</td>
<td>111506</td>
<td>223012</td>
<td>446024</td>
</tr>
<tr>
<td>16</td>
<td>21</td>
<td>413</td>
<td>3823</td>
<td>3328</td>
<td>16632</td>
<td>33264</td>
<td>66528</td>
<td>133056</td>
<td>266112</td>
</tr>
<tr>
<td>17</td>
<td>6</td>
<td>203</td>
<td>2472</td>
<td>16351</td>
<td>32702</td>
<td>65404</td>
<td>130808</td>
<td>261616</td>
<td>523232</td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>84</td>
<td>1128</td>
<td>4866</td>
<td>97328</td>
<td>194656</td>
<td>389312</td>
<td>778624</td>
<td>1557248</td>
</tr>
<tr>
<td>19</td>
<td>28</td>
<td>728</td>
<td>81451</td>
<td>162902</td>
<td>325804</td>
<td>651608</td>
<td>1293216</td>
<td>2586432</td>
<td>5172864</td>
</tr>
<tr>
<td>20</td>
<td>7</td>
<td>322</td>
<td>4950</td>
<td>9900</td>
<td>19800</td>
<td>39600</td>
<td>79200</td>
<td>158400</td>
<td>316800</td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>120</td>
<td>2598</td>
<td>5196</td>
<td>10392</td>
<td>20784</td>
<td>41568</td>
<td>83136</td>
<td>166272</td>
</tr>
<tr>
<td>22</td>
<td>36</td>
<td>1206</td>
<td>165</td>
<td>3306</td>
<td>6612</td>
<td>13224</td>
<td>26448</td>
<td>52896</td>
<td>105792</td>
</tr>
<tr>
<td>23</td>
<td>8</td>
<td>165</td>
<td>45</td>
<td>90</td>
<td>180</td>
<td>360</td>
<td>720</td>
<td>1440</td>
<td>2880</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>165</td>
<td>9</td>
<td>45</td>
<td>90</td>
<td>180</td>
<td>360</td>
<td>720</td>
<td>1440</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>45</td>
<td>9</td>
<td>45</td>
<td>90</td>
<td>180</td>
<td>360</td>
<td>720</td>
<td>1440</td>
</tr>
<tr>
<td>26</td>
<td>9</td>
<td>1</td>
<td>45</td>
<td>9</td>
<td>45</td>
<td>90</td>
<td>180</td>
<td>360</td>
<td>720</td>
</tr>
<tr>
<td>27</td>
<td>1</td>
<td>1</td>
<td>45</td>
<td>9</td>
<td>45</td>
<td>90</td>
<td>180</td>
<td>360</td>
<td>720</td>
</tr>
</tbody>
</table>
Table C-4 The path function $g_k^r(n)$ for $k = 5$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>36</td>
<td>45</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
<td>84</td>
<td>120</td>
<td>165</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>15</td>
<td>35</td>
<td>70</td>
<td>126</td>
<td>210</td>
<td>330</td>
<td>495</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>6</td>
<td>18</td>
<td>52</td>
<td>121</td>
<td>246</td>
<td>455</td>
<td>784</td>
<td>1278</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>7</td>
<td>19</td>
<td>68</td>
<td>185</td>
<td>426</td>
<td>875</td>
<td>1652</td>
<td>2922</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>8</td>
<td>20</td>
<td>80</td>
<td>235</td>
<td>666</td>
<td>1520</td>
<td>3144</td>
<td>6030</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>9</td>
<td>21</td>
<td>85</td>
<td>320</td>
<td>951</td>
<td>2115</td>
<td>4285</td>
<td>8570</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>10</td>
<td>22</td>
<td>90</td>
<td>365</td>
<td>991</td>
<td>2322</td>
<td>4646</td>
<td>9292</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>11</td>
<td>23</td>
<td>96</td>
<td>420</td>
<td>1056</td>
<td>2474</td>
<td>4948</td>
<td>9896</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>12</td>
<td>24</td>
<td>101</td>
<td>481</td>
<td>1181</td>
<td>2754</td>
<td>5508</td>
<td>10916</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>13</td>
<td>25</td>
<td>106</td>
<td>546</td>
<td>1306</td>
<td>3134</td>
<td>6368</td>
<td>12736</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>14</td>
<td>26</td>
<td>111</td>
<td>615</td>
<td>1431</td>
<td>3554</td>
<td>7108</td>
<td>14216</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>15</td>
<td>27</td>
<td>116</td>
<td>688</td>
<td>1556</td>
<td>3994</td>
<td>7992</td>
<td>15984</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>16</td>
<td>28</td>
<td>121</td>
<td>765</td>
<td>1681</td>
<td>4444</td>
<td>9088</td>
<td>18176</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>17</td>
<td>29</td>
<td>126</td>
<td>846</td>
<td>1806</td>
<td>4904</td>
<td>9968</td>
<td>19936</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>18</td>
<td>30</td>
<td>131</td>
<td>931</td>
<td>1931</td>
<td>5374</td>
<td>10928</td>
<td>21856</td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>19</td>
<td>31</td>
<td>136</td>
<td>1019</td>
<td>2056</td>
<td>5854</td>
<td>11888</td>
<td>23776</td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>20</td>
<td>32</td>
<td>141</td>
<td>1109</td>
<td>2181</td>
<td>6344</td>
<td>12848</td>
<td>25664</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>21</td>
<td>33</td>
<td>146</td>
<td>1199</td>
<td>2306</td>
<td>6844</td>
<td>13808</td>
<td>27552</td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>22</td>
<td>34</td>
<td>151</td>
<td>1291</td>
<td>2431</td>
<td>7354</td>
<td>14768</td>
<td>29440</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>23</td>
<td>35</td>
<td>156</td>
<td>1384</td>
<td>2556</td>
<td>7874</td>
<td>15728</td>
<td>31328</td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td>24</td>
<td>36</td>
<td>161</td>
<td>1478</td>
<td>2681</td>
<td>8404</td>
<td>16688</td>
<td>33216</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>25</td>
<td>37</td>
<td>166</td>
<td>1573</td>
<td>2806</td>
<td>8934</td>
<td>17648</td>
<td>35104</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>26</td>
<td>38</td>
<td>171</td>
<td>1668</td>
<td>2931</td>
<td>9464</td>
<td>18608</td>
<td>37092</td>
</tr>
<tr>
<td>26</td>
<td>1</td>
<td>27</td>
<td>39</td>
<td>176</td>
<td>1765</td>
<td>3056</td>
<td>10004</td>
<td>19568</td>
<td>39080</td>
</tr>
<tr>
<td>27</td>
<td>1</td>
<td>28</td>
<td>40</td>
<td>181</td>
<td>1862</td>
<td>3181</td>
<td>10534</td>
<td>20528</td>
<td>41068</td>
</tr>
<tr>
<td>28</td>
<td>1</td>
<td>29</td>
<td>41</td>
<td>186</td>
<td>1960</td>
<td>3306</td>
<td>11064</td>
<td>21488</td>
<td>43056</td>
</tr>
<tr>
<td>29</td>
<td>1</td>
<td>30</td>
<td>42</td>
<td>191</td>
<td>2059</td>
<td>3431</td>
<td>11594</td>
<td>22448</td>
<td>45044</td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td>31</td>
<td>43</td>
<td>196</td>
<td>2158</td>
<td>3556</td>
<td>12124</td>
<td>23408</td>
<td>47032</td>
</tr>
<tr>
<td>31</td>
<td>1</td>
<td>32</td>
<td>44</td>
<td>201</td>
<td>2258</td>
<td>3681</td>
<td>12654</td>
<td>24368</td>
<td>49020</td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td>33</td>
<td>45</td>
<td>206</td>
<td>2358</td>
<td>3806</td>
<td>13184</td>
<td>25328</td>
<td>51008</td>
</tr>
<tr>
<td>33</td>
<td>1</td>
<td>34</td>
<td>46</td>
<td>211</td>
<td>2458</td>
<td>3931</td>
<td>13714</td>
<td>26288</td>
<td>53006</td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>35</td>
<td>47</td>
<td>216</td>
<td>2558</td>
<td>4056</td>
<td>14244</td>
<td>27248</td>
<td>55004</td>
</tr>
<tr>
<td>35</td>
<td>1</td>
<td>36</td>
<td>48</td>
<td>221</td>
<td>2658</td>
<td>4181</td>
<td>14774</td>
<td>28208</td>
<td>57002</td>
</tr>
<tr>
<td>36</td>
<td>1</td>
<td>37</td>
<td>49</td>
<td>226</td>
<td>2758</td>
<td>4306</td>
<td>15304</td>
<td>29168</td>
<td>59000</td>
</tr>
</tbody>
</table>
Table C-5 The path function $g_k^K(l)$ for $K = 6$.

<table>
<thead>
<tr>
<th>l</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>36</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>36</td>
<td>330</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
<td>81</td>
<td>120</td>
<td>792</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>15</td>
<td>35</td>
<td>70</td>
<td>126</td>
<td>210</td>
<td>330</td>
<td>1708</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>21</td>
<td>56</td>
<td>126</td>
<td>252</td>
<td>462</td>
<td>792</td>
<td>3368</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>27</td>
<td>104</td>
<td>305</td>
<td>756</td>
<td>1667</td>
<td>3368</td>
<td>10880</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>32</td>
<td>125</td>
<td>420</td>
<td>1161</td>
<td>2307</td>
<td>6147</td>
<td>10880</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>25</td>
<td>140</td>
<td>510</td>
<td>1666</td>
<td>4117</td>
<td>10880</td>
<td>6147</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>21</td>
<td>146</td>
<td>651</td>
<td>2217</td>
<td>6538</td>
<td>16808</td>
<td>50288</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>15</td>
<td>110</td>
<td>735</td>
<td>2556</td>
<td>9114</td>
<td>25488</td>
<td>50288</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>10</td>
<td>125</td>
<td>780</td>
<td>3431</td>
<td>12117</td>
<td>36688</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>6</td>
<td>104</td>
<td>780</td>
<td>3906</td>
<td>15267</td>
<td>50288</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>14</td>
<td>3</td>
<td>80</td>
<td>735</td>
<td>4221</td>
<td>18327</td>
<td>65808</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>15</td>
<td>1</td>
<td>56</td>
<td>651</td>
<td>1432</td>
<td>20993</td>
<td>82384</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>16</td>
<td>35</td>
<td>540</td>
<td>1422</td>
<td>22967</td>
<td>98813</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>20</td>
<td>420</td>
<td>3906</td>
<td>22017</td>
<td>113688</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>18</td>
<td>4</td>
<td>205</td>
<td>3431</td>
<td>22017</td>
<td>125388</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>19</td>
<td>1</td>
<td>126</td>
<td>2217</td>
<td>20993</td>
<td>135954</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>70</td>
<td>1666</td>
<td>18327</td>
<td>133283</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>35</td>
<td>1161</td>
<td>15267</td>
<td>125588</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>22</td>
<td>15</td>
<td>756</td>
<td>12117</td>
<td>113688</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>23</td>
<td>5</td>
<td>456</td>
<td>9142</td>
<td>98813</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>24</td>
<td>1</td>
<td>252</td>
<td>6538</td>
<td>82384</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>25</td>
<td>126</td>
<td>4141</td>
<td>65808</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>26</td>
<td>56</td>
<td>2807</td>
<td>50288</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>27</td>
<td>21</td>
<td>1667</td>
<td>36688</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>28</td>
<td>6</td>
<td>917</td>
<td>25488</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>29</td>
<td>1</td>
<td>462</td>
<td>16808</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>30</td>
<td>210</td>
<td>814</td>
<td>3368</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>31</td>
<td>84</td>
<td>6247</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>32</td>
<td>28</td>
<td>7</td>
<td>1708</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>33</td>
<td>7</td>
<td>1</td>
<td>792</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>34</td>
<td>36</td>
<td>8</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table C-6 The path function $g^7_k(l)$ for $K = 7.$

<table>
<thead>
<tr>
<th>$l$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
<td>81</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>15</td>
<td>35</td>
<td>70</td>
<td>126</td>
<td>210</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>6</td>
<td>21</td>
<td>56</td>
<td>126</td>
<td>252</td>
<td>462</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>7</td>
<td>28</td>
<td>84</td>
<td>210</td>
<td>462</td>
<td>924</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>8</td>
<td>33</td>
<td>116</td>
<td>325</td>
<td>786</td>
<td>1709</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>9</td>
<td>36</td>
<td>119</td>
<td>470</td>
<td>1251</td>
<td>2955</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>10</td>
<td>37</td>
<td>180</td>
<td>640</td>
<td>1876</td>
<td>4809</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>11</td>
<td>36</td>
<td>206</td>
<td>826</td>
<td>2567</td>
<td>7120</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>33</td>
<td>224</td>
<td>1015</td>
<td>3612</td>
<td>10906</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>28</td>
<td>231</td>
<td>1190</td>
<td>4676</td>
<td>15330</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>224</td>
<td>1330</td>
<td>5796</td>
<td>20664</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>206</td>
<td>1420</td>
<td>6852</td>
<td>26769</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>180</td>
<td>1451</td>
<td>7872</td>
<td>33390</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>6</td>
<td>149</td>
<td>1420</td>
<td>8652</td>
<td>40166</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>3</td>
<td>116</td>
<td>1330</td>
<td>9156</td>
<td>46655</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>84</td>
<td>1190</td>
<td>9331</td>
<td>52374</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>2</td>
<td>56</td>
<td>1015</td>
<td>9156</td>
<td>56854</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>35</td>
<td>826</td>
<td>8652</td>
<td>59710</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>20</td>
<td>610</td>
<td>7072</td>
<td>60591</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>10</td>
<td>470</td>
<td>6891</td>
<td>59710</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>4</td>
<td>325</td>
<td>5796</td>
<td>56854</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>210</td>
<td>4676</td>
<td>52374</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>126</td>
<td>3612</td>
<td>46655</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>70</td>
<td>2667</td>
<td>40166</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>35</td>
<td>1876</td>
<td>33390</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>15</td>
<td>1251</td>
<td>28769</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>5</td>
<td>786</td>
<td>20664</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td>462</td>
<td>15330</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>252</td>
<td>10906</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>126</td>
<td>7420</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>56</td>
<td>4809</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>21</td>
<td>2954</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>6</td>
<td>1709</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>1</td>
<td>954</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>62</td>
<td>162</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>210</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>84</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table C-7 The path function $g_k^K(l)$ for $K = 8$.

<table>
<thead>
<tr>
<th>$g$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>15</td>
<td>35</td>
<td>70</td>
<td>126</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>6</td>
<td>21</td>
<td>56</td>
<td>126</td>
<td>252</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>7</td>
<td>28</td>
<td>84</td>
<td>210</td>
<td>462</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>8</td>
<td>36</td>
<td>120</td>
<td>330</td>
<td>792</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>9</td>
<td>45</td>
<td>161</td>
<td>460</td>
<td>1281</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>10</td>
<td>54</td>
<td>216</td>
<td>690</td>
<td>1966</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>11</td>
<td>64</td>
<td>285</td>
<td>926</td>
<td>2877</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>12</td>
<td>74</td>
<td>364</td>
<td>1190</td>
<td>3032</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>13</td>
<td>84</td>
<td>453</td>
<td>1470</td>
<td>3936</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>14</td>
<td>94</td>
<td>552</td>
<td>1850</td>
<td>5006</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>15</td>
<td>104</td>
<td>661</td>
<td>2250</td>
<td>6062</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>16</td>
<td>114</td>
<td>778</td>
<td>2650</td>
<td>7203</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>17</td>
<td>124</td>
<td>897</td>
<td>3050</td>
<td>8406</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>18</td>
<td>134</td>
<td>1017</td>
<td>3450</td>
<td>9602</td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>19</td>
<td>144</td>
<td>1138</td>
<td>3850</td>
<td>10803</td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>20</td>
<td>154</td>
<td>1260</td>
<td>4250</td>
<td>12004</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>21</td>
<td>164</td>
<td>1382</td>
<td>4650</td>
<td>13206</td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>22</td>
<td>174</td>
<td>1508</td>
<td>5050</td>
<td>14408</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>23</td>
<td>184</td>
<td>1636</td>
<td>5450</td>
<td>15609</td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td>24</td>
<td>194</td>
<td>1766</td>
<td>5850</td>
<td>16809</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>25</td>
<td>204</td>
<td>1898</td>
<td>6250</td>
<td>18008</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>26</td>
<td>214</td>
<td>2032</td>
<td>6650</td>
<td>19207</td>
</tr>
<tr>
<td>26</td>
<td>1</td>
<td>27</td>
<td>224</td>
<td>2168</td>
<td>7050</td>
<td>20406</td>
</tr>
<tr>
<td>27</td>
<td>1</td>
<td>28</td>
<td>234</td>
<td>2306</td>
<td>7450</td>
<td>21607</td>
</tr>
<tr>
<td>28</td>
<td>1</td>
<td>29</td>
<td>244</td>
<td>2446</td>
<td>7850</td>
<td>22806</td>
</tr>
<tr>
<td>29</td>
<td>1</td>
<td>30</td>
<td>254</td>
<td>2590</td>
<td>8250</td>
<td>24005</td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td>31</td>
<td>264</td>
<td>2742</td>
<td>8650</td>
<td>25204</td>
</tr>
<tr>
<td>31</td>
<td>1</td>
<td>32</td>
<td>274</td>
<td>2896</td>
<td>9050</td>
<td>26403</td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td>33</td>
<td>284</td>
<td>3052</td>
<td>9450</td>
<td>27602</td>
</tr>
<tr>
<td>33</td>
<td>1</td>
<td>34</td>
<td>294</td>
<td>3210</td>
<td>9850</td>
<td>28802</td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>35</td>
<td>304</td>
<td>3370</td>
<td>10250</td>
<td>29992</td>
</tr>
<tr>
<td>35</td>
<td>1</td>
<td>36</td>
<td>314</td>
<td>3532</td>
<td>10650</td>
<td>31192</td>
</tr>
<tr>
<td>36</td>
<td>1</td>
<td>37</td>
<td>324</td>
<td>3704</td>
<td>11050</td>
<td>32392</td>
</tr>
<tr>
<td>37</td>
<td>1</td>
<td>38</td>
<td>334</td>
<td>3880</td>
<td>11450</td>
<td>33592</td>
</tr>
<tr>
<td>38</td>
<td>1</td>
<td>39</td>
<td>344</td>
<td>4060</td>
<td>11850</td>
<td>34792</td>
</tr>
<tr>
<td>39</td>
<td>1</td>
<td>40</td>
<td>354</td>
<td>4242</td>
<td>12250</td>
<td>35992</td>
</tr>
<tr>
<td>40</td>
<td>1</td>
<td>41</td>
<td>364</td>
<td>4428</td>
<td>12650</td>
<td>37192</td>
</tr>
<tr>
<td>41</td>
<td>1</td>
<td>42</td>
<td>374</td>
<td>4616</td>
<td>13050</td>
<td>38392</td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td>43</td>
<td>384</td>
<td>4808</td>
<td>13450</td>
<td>39592</td>
</tr>
</tbody>
</table>
APPENDIX D

SUBROUTINE LISTING

The following pages give a FORTRAN listing of the non-linear programming subroutine FMINOD (Function Minimization Without Derivatives). This subroutine implements the simplex algorithm suggested by Nelder and Mead [25].
SUBROUTINE FMWOD(X,F,N,PRTURB, EPS, LIMIT)

*****************************************************************************
* SUBROUTINE TO MINIMIZE A GENERAL FUNCTION OF A NUMBER OF VARIABLES WITHOUT CALCULATING DERIVATIVES *
*****************************************************************************

THIS SUBROUTINE CALCULATES THE MINIMUM OF A GENERAL FUNCTION OF A NUMBER OF VARIABLES. THE CALCULATION OF PARTIAL DERIVATIVES IS NOT REQUIRED AND IT IS NOT EVEN NECESSARY THAT THE FUNCTION BE DIFFERENTIABLE. THE SIMPLEX METHOD OF NELDER AND MEAD WHICH IS DESCRIBED IN THE (BRITISH) COMPUTER JOURNAL, VOL 7, 1964, IS UTILIZED.

USAGE IS AS FOLLOWS

X - N-VECTOR OF INDEPENDENT VARIABLES. INITIALLY, IT IS THE USERS BEST GUESS OF THE MINIMUM. SUBROUTINE FMWOD RETURNS WITH THE VALUE OF X CORRESPONDING TO THE MINIMUM.

F - VALUE OF THE FUNCTION AT THE MINIMUM.

N - THE NUMBER OF INDEPENDENT VARIABLES. N.LE.50

PRTURB- N-VECTOR REPRESENTING A SET OF SUGGESTED PERTURBATIONS FROM THE INITIAL VALUE OF X.

EPS - TEST VALUE REPRESENTING THE EXPECTED ABSOLUTE ERROR. EPS IS USED IN THE TESTING FOR CONVERGENCE.

LIMIT - INITIALLY, THE MAXIMUM NUMBER OF ITERATIONS TO BE ALLOWED. SUBROUTINE RETURNS WITH LIMIT = THE ACTUAL NUMBER OF ITERATIONS USED IF THE SUBROUTINE CONVERGED. LIMIT = 0, OTHERWISE.

SUBROUTINE FUNCT(X,F) - SUBROUTINE WHICH EVALUATES THE FUNCTION F AT THE POINT X. THIS SUBROUTINE IS USED REPEATEDLY BY FMWOD.

DIMENSION S(51,51), X(50), PRTURB(50), SC(50)

FN = FLOAT(N)
NP1 = N + 1
C SET UP INITIAL SIMPLEX
   DO 4 I=1,N
   DO 3 J=1,NP1
   3 S(I,J) = X(I)
   4 S(I,I) = X(I) + PRTURB(I)
   DO 7 J=1,NP1
   DO 6 I=1,N
   6 X(I) = S(I,J)
   CALL FUNCT(X,F)
   7 S(NP1,J) = F
C START ITERATION
   DO 99 NUM=1,LIMIT
   C FIND THE HIGH AND LOW POINTS
   FL = S(NP1,1)
   FH = FL
   FH2 = -1.E+30
   JH = 1
   JL = 1
   DO 13 J=2,NP1
   BURP = S(NP1,J)
   IF(FH.GE.BURP) GO TO 11
   FH2 = FH
   FH = BURP
   JH = J
   GO TO 13
   11 IF(FH2.GE.BURP) GO TO 12
   FH2 = BURP
   12 IF(FL.LE.BURP) GO TO 13
   FL = BURP
   JL = J
   13 CONTINUE
C CHECK FOR CONVERGENCE
   IF(FH-FL.LT.EPS) GO TO 100
C FIND THE CENTROID OF THE SIMPLEX
   DO 16 I=1,N
   SUM = 0.
   DO 15 J=1,NP1
   15 SUM = SUM + S(I,J)
   16 SC(I) = (SUM - S(I,JH))/FN
C TRY A REFLECTION
   DO 19 I=1,N
   19 X(I) = 2.*SC(I) - S(I,JH)
   CALL FUNCT(X,F)
   IF(F.LT.FL) GO TO 50
   IF(F.LT.FH2) GO TO 60
   IF(F.GE.FH) GO TO 25
C REFLECTION WAS A PARTIAL SUCCESS. STORE X.
   DO 21 I=1,N
   21 S(I,JH) = X(I)
\[ S(NP1, JH) = F \]
\[ FH = F \]

C TRY A CONTRACTION
25 DO 26 I=1,N
26 \[ X(I) = (S(I,JH) + SC(I))/2.0 \]
CALL FUNCT(X,F)
IF(F.LT.FH) GO TO 60

C FAILED CONTRACTION. SHRINK SIMPLEX ABOUT ITS LOW POINT
DO 32 J=1,NP1
IF(J.EQ.JL) GO TO 32
DO 31 I=1,N
BURP = (S(1,J) + S(I,JL))/2.0
\[ S(I,J) = BURP \]
\[ X(I) = BURP \]
CALL FUNCT(X,F)
\[ S(NP1,J) = F \]
31 CONTINUE
32 CONTINUE GO TO 99

C VERY SUCCESSFUL REFLECTION. STORE X AND THEN TRY AN EXPANSION
50 DO 51 I=1,N
\[ S(I,JH) = X(I) \]
51 \[ X(I) = 2.*X(I) - SC(I) \]
\[ S(NP1,JH) = F \]
\[ FL = F \]
CALL FUNCT(X,F)
IF(F.GE.FL) GO TO 99

C SUCCESSFUL REFLECTION, CONTRACTION, OR EXPANSION. STORE X.
60 DO 61 I=1,N
61 \[ S(I,JH) = X(I) \]
\[ S(NP1,JH) = F \]
99 CONTINUE

C ROUTINE DID NOT CONVERGE
LIMIT = 0
GO TO 101

C ROUTINE CONVERGED. STORE ANSWER AND RETURN
100 LIMIT = NUM - 1
101 DO 102 I=1,N
102 \[ X(I) = S(I,JL) \]
\[ F = S(NP1,JL) \]
RETURN
END
REFERENCES


11. H. E. Scarf and L. S. Shapley, "Games with Information Lag," 

in *Annals of Mathematics Studies*, No. 39. Princeton, New Jersey: 

Proceedings of a Conference under the Auspices of the NATO Scientific 
Affairs Committee in Toulon 1964*. New York: American Elsevier, 
1966, pp. 276-284.


15. J. L. Speyer, "A Stochastic Differential Game with Controllable 

16. A. F. Bulfer, "Optimal Tracking and Future Prediction of an Evading, 

17. A. F. Bulfer and H. Hemami, "Tracking and Future Prediction of an 


